A Comprehensive Study of Bengali Fake News Detection Using Machine Learning, Deep Learning, and Transformer-Based Ensemble Approaches

by

Fariha Afrin
ID: CSE2203027160

Pial Hasan
ID: CSE2203027158

Md Afridi Hossain
ID: CSE2203027157

Md Juwel Rana
ID: CSE2203027115

Momena Mahmud Muna
ID: CSE2203027099

Supervised by
Tasnia Haque Kheya

Submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SONARGAON UNIVERSITY (SU)
September 2026

APPROVAL

The thesis titled "**A Comprehensive Study of Bengali Fake News Detection Using Machine Learning, Deep Learning, and Transformer-Based Ensemble Approaches**" submitted by Fariha Afrin (CSE2203027160), Pial Hasan (CSE2203027158), Md Afridi Hossain (CSE2203027157), Md Juwel Rana (CSE2203027115) and Momena Mahmud Muna(CSE2203027099) to the Department of Computer Science and Engineering, Sonargaon University (SU), has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

Board of Examiners

Tasnia Haque Kheya **Supervisor**

Lecturer
Department of Computer Science and Engineering
Sonargaon University (SU)

(Examiner Name and Signature) **Examiner 1**

Department of Computer Science and Engineering
Sonargaon University (SU)

(Examiner Name and Signature) **Examiner 2**

Department of Computer Science and Engineering
Sonargaon University (SU)

(Examiner Name and Signature) **Examiner 3**

Department of Computer Science and Engineering
Sonargaon University (SU)

DECLARATION

We, hereby, declare that the work presented in this report is the outcome of the investigation performed by us under the supervision of **Tasnia Haque Kheya, Lecturer,** Department of Computer Science and Engineering, Sonargaon University, Dhaka, Bangladesh. We reaffirm that no part of this thesis has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned

(Tasnia Haque Kheya)
Supervisor

Signature

Fariha Afrin
ID: CSE2203027160

Pial Hasan
ID: CSE2203027158

Md Afridi Hossain
ID: CSE2203027157

Md. Juwel Rana
ID: CSE2203027115

Momena Mahmud Muna
ID: CSE2203027099

ABSTRACT

The rapid spread of misinformation on social media and online news portals makes automated fake news detection a critical research problem, especially for low-resource languages like Bengali. This thesis presents a leakage-aware pipeline that compares ML, DL, and Transformer-based Ensemble models on the same dataset to identify the most accurate and reliable approach. Al though many fake news detection studies already exist, most focus on English or use only one type of model. In contrast, this research focuses on Bengali and compares Machine Learning, Deep Learning, and Transformer-based Ensemble models on the same dataset. The main objective is to identify the most accurate and reliable approach for detecting Bengali fake news by collecting and preprocessing Bengali news, training different AI models, evaluating their performance, and selecting the best-performing model. A total of 27 models, including classical ML, DL, BanglaBERT, XLM-RoBERTa, and eight ensemble strategies, were trained using TF-IDF and stylometric features. SHAP, LIME, and attention analysis confirmed that the models relied on genuine content rather than artifacts. By avoiding data leakage, the results provide a more trustworthy performance estimate, while dataset size and label noise, rather than model architecture, remain the main limitations. Among all 27 evaluated models, the Soft-Voting Ensemble achieved the best overall performance with 87.09% accuracy, 88.79% F1-score, and 93.96% ROC-AUC. It outperformed all individual models by combining the strengths of diverse Machine Learning, Deep Learning, and Transformer-based models. In complex situations, the model predicts based on learned patterns. Since no AI model is 100% accurate, it should be used as a decision-support tool. Transformer models improve context understanding, and Ensemble methods combine multiple models to improve accuracy, making Bengali fake news detection more effective.

ACKNOWLEDGEMENT

At the very outset, I would like to express my deepest gratitude to the Almighty for granting me the strength, patience, and clarity of mind required to complete this research within the scheduled time.

I am profoundly grateful to my supervisor of **Tasnia Haque Kheya**, Lecturer, Department of Computer Science and Engineering, Sonargaon University (SU), for the continuous guidance, insightful suggestions, and constant encouragement extended throughout this research. Their expertise, patience in reviewing successive iterations of the experimental pipeline, and constructive feedback on methodological rigour particularly regarding data-leakage prevention were invaluable at every stage of this work.

I would also like to convey my sincere gratitude to **Prof. Bulbul Ahamed**, Head, Department of Computer Science and Engineering, Sonargaon University (SU), and to all the respected faculty members of the department, for their continuous motivation, academic support, and for creating an environment in which independent research of this kind could be pursued.

Finally, I owe my deepest personal gratitude to my parents and family members for their unconditional love, patience, and unwavering support throughout my academic journey, without which this work would not have been possible.

LIST OF ABBREVIATIONS

Abbreviation	Full Form
AB	AdaBoost
AI	Artificial Intelligence
AMP / FP16	Automatic Mixed Precision / 16-bit Floating Point
API	Application Programming Interface
AUC	Area Under the Curve
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
BoW	Bag of Words
CatBoost	Categorical Boosting
CNN	Convolutional Neural Network
CSV	Comma-Separated Values
CV	Cross-Validation
DL	Deep Learning
EDA	Exploratory Data Analysis
ET	Extra Trees
GB	Gradient Boosting
GPU / CPU	Graphics Processing Unit / Central Processing Unit
GRU	Gated Recurrent Unit
HTML	HyperText Markup Language
JS	JavaScript
KNN	K-Nearest Neighbors
LightGBM	Light Gradient Boosting Machine
LIME	Local Interpretable Model-agnostic Explanations
LR	Logistic Regression
LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
mBERT	Multilingual BERT
ML	Machine Learning
NB	Naive Bayes

NFC	Unicode Normalization Form Canonical Composition
NLP	Natural Language Processing
OOF	Out-of-Fold (prediction)
PR (curve)	Precision–Recall (curve)
RBF	Radial Basis Function (SVM kernel)
RF	Random Forest
ROC	Receiver Operating Characteristic
RNN	Recurrent Neural Network
SHAP	SHapley Additive exPlanations
SMOTE	Synthetic Minority Over-sampling Technique
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TF-IDF	Term Frequency – Inverse Document Frequency
TN / FN / TP / FP	True Negative / False Negative / True Positive / False Positive
TTS	Train–Test Split
URL	Uniform Resource Locator
XGB / XGBoost	Extreme Gradient Boosting
XLM-R	Cross-lingual Language Model – RoBERTa
ZWJ / ZWNJ	Zero-Width Joiner / Zero-Width Non-Joiner

TABLE OF CONTENTS

Title			Page No.
DECLARATION			**iii**
ABSTRACT			**iv**
ACKNOWLEDGEMENT			**v**
LIST OF ABBREVIATIONS …….....			**vi-vii**
CHAPTER 1			**1 – 17**
INTRODUCTION			
1.1	Background of the Study		1
	1.1.1	The Digital Revolution in Bangladesh	1
	1.1.2	Rise of Bengali Content on Social Media	2
	1.1.3	Definition and Types of Fake News	2
	1.1.4	Real-world Examples of Bangla Fake News Incidents	3
	1.1.5	Consequences of Misinformation in Bengali-Speaking Communities	4
	1.1.6	Need for Automated Fake News Detection in Bangla	5
1.2	Problem Statement		7
	1.2.1	Ineffectiveness of English-based Detection Systems for Bangla	7
	1.2.2	Linguistic Complexities of Bengali Language	8
	1.2.3	Scarcity of Labeled Bangla Fake News Datasets	9
	1.2.4	Absence of Real-time Detection Tools for Bangla Content	10
	1.2.5	Social and Political Impact of Unchecked Fake News in Bangladesh	10
1.3	Objectives of the Study		11
	1.3.1	Objective 1: Dataset Construction	11
	1.3.2	Objective 2: Bangla Text Preprocessing Pipeline	12
	1.3.3	Objective 3: Feature Extraction from Bangla Text	12
	1.3.4	Objective 4: Model Development and Training	12
	1.3.5	Objective 5: Performance Evaluation and Comparative Analysis	12
1.4	Research Questions		13
	1.4.1	Effective Preprocessing for Bangla Text	13
	1.4.2	Optimal Feature Extraction Technique	13
	1.4.3	Deep Learning vs Traditional ML Performance	13
	1.4.4	Effectiveness of Bangla-BERT	13
	1.4.5	Linguistic Cues for Fake News Identification	14
1.5	Significance of the Research		14
	1.5.1	Academic Contribution to Bengali NLP	14
	1.5.2	Social Impact on Bangladeshi Society	14

1.5.3 Technological Contribution for News Agencies 14
1.5.4 Economic and Commercial Potential 14
1.5.5 Support for Fact-Checking Organizations 15
1.6 Scope and Limitations 15
1.6.1 Scope of the Research 15
1.6.1.1 Text-based Bangla Content Only 15
1.6.1.2 Supervised Learning Approach 15
1.6.1.3 News Articles and Social Media Posts 15
1.6.1.4 Offline Classification System 15
1.6.2 Limitations of the Research 15
1.6.2.1 Limited Dataset Size 15
1.6.2.2 Inability to Detect Image/Video Fake News 16
1.6.2.3 Challenges with Bengali Dialects 16
1.6.2.4 No Real-time Processing Capability 16
1.6.2.5 Poor Performance on “Benglish” (Mixed Language) 16
1.7 Organization of the Thesis 16

CHAPTER 2 **18 – 29**
LITERATURE REVIEW

2.1 Introduction 18
2.2 Fake News Detection: A Global Perspective 18
2.2.1 Traditional Approaches (Rule-based and Linguistic) 18
2.2.2 Machine Learning-based Detection Approaches 19
2.2.3 Deep Learning-based Detection Approaches 19
2.2.4 Transformer and Pretrained Language Model Approaches 19
2.3 Fake News Detection in Low-Resource Languages 20
2.4 Existing Research on Bengali Fake News Detection 20
2.4.1 Dataset-oriented Studies 20
2.4.2 Feature Engineering-based Studies 21
2.4.3 Deep Learning and BanglaBERT-based Studies 21
2.5 Bengali Natural Language Processing: Tools and Challenges 22
2.5.1 Unicode Normalization and Text Encoding Issues 22
2.5.2 Absence of Reliable Stemmers/Lemmatizers 22
2.5.3 Available Bengali Corpora and Pretrained Models 23
2.6 Feature Extraction Techniques in Text Classification 23
2.6.1 Statistical Features (TF-IDF, N-grams) 23
2.6.2 Handcrafted Linguistic and Stylistic Features 23
2.6.3 Word Embeddings (Word2Vec, FastText) 23
2.6.4 Contextual Embeddings (BERT-based Models) 24
2.7 Ensemble Learning in Text Classification 24

2.7.1 Voting-based Ensembles 24
2.7.2 Blending and Stacking Techniques 25
2.8 Comparative Summary of Related Works 25
2.9 Research Gap Identification 28
2.10 Summary 28

CHAPTER 3 **30 – 48**
RESEARCH METHODOLOGY
3.1 Introduction 30
3.2 Methodology Overview 30
3.3 Dataset Construction 31
3.3.1 Data Sources and Collection Strategy 31
3.3.2 Category-wise Data Distribution 31
3.3.3 Annotation Protocol and Label Agreement 31
3.3.4 Data Quality Assessment 31
3.3.5 Deduplication and Label Conflict Resolution 33
3.3.6 Final Dataset Description and Class Distribution 34
3.4 Bangla Text Preprocessing Pipeline 34
3.4.1 Unicode Normalization (NFC) 35
3.4.2 Noise and Artifact Removal (HTML, URLs, Ads, Scraping Tags) 35
3.4.3 Outlet-marker and Shortcut-learning Prevention 35
3.4.4 Emoji, Hashtag, and Special Character Handling 35
3.4.5 Text Normalization (Elongation, Punctuation Handling) 36
3.4.6 Character Set Management 36
3.4.7 Handling of Stemming/Lemmatization Limitations 36
3.5 Feature Extraction Framework 36
3.5.1 Word-level TF-IDF Features 36
3.5.2 Character-level N-gram Features 37
3.5.3 Handcrafted Meta-Features 37
3.5.3.1 Length-based Features 38
3.5.3.2 Lexical Diversity and Entropy Features 38
3.5.3.3 Stylistic Features (Punctuation, Sensationalism Markers) 38
3.5.3.4 Journalistic Rigor Features 38
3.5.3.5 Digit Density Features 38
3.5.4 Word2Vec Embeddings 38
3.5.5 BanglaBERT Contextual Embeddings 39
3.5.6 Feature Integration and Scaling 39
3.6 Machine Learning Models 39
3.6.1 Logistic Regression 40

3.6.2 Support Vector Machines (Linear and RBF) 40
3.6.3 Naive Bayes 40
3.6.4 Random Forest and Extra Trees 40
3.6.5 AdaBoost and Gradient Boosting 41
3.6.6 XGBoost, LightGBM, and CatBoost 41
3.7 Deep Learning Models 41
3.7.1 TextCNN 41
3.7.2 BiLSTM and GRU 41
3.7.3 CNN-BiLSTM Hybrid Architecture 42
3.7.4 BiLSTM with Attention Mechanism 42
3.8 Transformer-based Models 42
3.8.1 BanglaBERT Fine-tuning 42
3.8.2 XLM-RoBERTa Fine-tuning 42
3.9 Ensemble Learning Methods 43
3.9.1 Soft Voting 43
3.9.2 Hard Voting 43
3.9.3 Weighted Voting (Dirichlet Optimization) 43
3.9.4 Blending 44
3.9.5 K-Fold Stacking 44
3.10 Experimental Setup 44
3.10.1 Train-Test-Validation Split Strategy 44
3.10.2 Hyperparameter Tuning and Cross-Validation 45
3.10.3 Programming Environment and Libraries 45
3.10.4 Hardware Configuration 45
3.11 Evaluation Metrics 45
3.11.1 Accuracy 45
3.11.2 Precision, Recall, and F1-Score 45
3.11.3 ROC-AUC 46
3.11.4 Confusion Matrix 46
3.12 Summary 46

CHAPTER 4 **49 – 87**
RESULTS AND DISCUSSION

4.1 Introduction 49
4.2 Exploratory Data Analysis 49
4.2.1 Class Distribution (Real vs Fake) 50
4.2.2 Category-wise Distribution of Articles 50
4.2.3 Text Length and Word Count Analysis 50
4.2.4 Word Frequency and N-gram Analysis 50
4.3 Performance of Traditional Machine Learning Models 51

4.4 Performance of Deep Learning Models 51
4.5 Performance of Transformer-based Models 53
4.6 Performance of Ensemble Models 53
4.7 Comparative Analysis of All Models 55
4.7.1 Comparison Table of Accuracy, Precision, Recall, F1-Score, ROC-AUC 55
4.7.2 Best-performing Model Identification 56
4.8 Feature Importance Analysis 57
4.9 Error Analysis and Misclassification Cases 60
4.10 Impact of Preprocessing on Model Performance 61
4.11 Discussion of Research Questions 62
4.11.1 RQ1 — Effective Preprocessing for Bangla Text 62
4.11.2 RQ2 — Optimal Feature Extraction Technique 62
4.11.3 RQ3 — Deep Learning vs Traditional ML Performance 62
4.11.4 RQ4 — Effectiveness of Bangla-BERT 62
4.11.5 RQ5 — Linguistic Cues for Fake News Identification 63
4.12 Visualization of Results 63
4.12.1 Confusion Matrices 63
4.12.2 ROC Curves 65
4.12.3 Model Comparison Charts 68
4.12.4 Feature Importance Plots 71
4.12.5 Additional Explainability, Robustness, and Efficiency Visualizations 71
4.13 Summary 86

CHAPTER 5 **88 – 91**
LIMITATIONS AND ETHICAL CONSIDERATIONS

5.1 Introduction 88
5.2 Limitations of the Study 88
5.2.1 Dataset Size and Domain Coverage 88
5.2.2 Text-only Modality Limitation 88
5.2.3 Dialectal and Benglish Variation Challenges 89
5.2.4 Computational Constraints of Transformer Models 89
5.3 Ethical Considerations 89
5.3.1 Data Privacy and Source Attribution 89
5.3.2 Bias and Fairness in Predictions 90
5.3.3 Responsible Use of Predictions 90
5.3.4 Transparency and Explainability 90
5.4 Summary 91

CHAPTER 6 **92 – 96**

CONCLUSION AND FUTURE WORK

6.1 Introduction 92
6.2 Summary of Research Work 92
6.3 Key Findings 92
6.3.1 Model Performance Insights 92
6.3.2 Feature Contribution Insights 93
6.3.3 Dataset-related Insights 93
6.4 Practical Implications 93
6.5 Contributions of the Study 93
6.6 Future Work 94
6.6.1 Larger and More Diverse Dataset Collection 94
6.6.2 Multimodal Detection (Text, Image, Video) 94
6.6.3 Real-time Prediction Systems 95
6.6.4 Bias Mitigation and Ethical AI 95
6.6.5 Longitudinal Performance Analysis 95
6.6.6 Deployment in Education Policy and as a Service 95
6.7 Concluding Remarks 96

REFERENCES ..... **97 – 99**

APPENDIX ..... **100 – 103**
Appendix A Sample Dataset Records 100
Appendix B Representative Code Excerpts 100
B.1 Outlet-marker and Scraping-artifact Shortcut Detection (Section 3.3.4) 100
B.2 Word- and Character-level TF-IDF Construction (Section 3.5.1–3.5.2, 3.5.6) 101
B.3 Diversity-Aware Automatic Base-Model Selection (Section 3.9) 101
B.4 Validation-only Weighted-Voting Optimisation (Section 3.9.3) 101
B.5 Bengali-Font-Aware Figure Rendering (referenced by Figures 4.9 and 4.12) 102
B.6 Handcrafted Meta-Feature Extraction (Section 3.5.3, Figure 4.15) 102
Appendix C Supplementary Hyperparameter Summary 103

LIST OF FIGURES

Figure No.	Title	Page No
Figure 3.1	Content-length distribution by class and category-wise article counts (exploratory check prior to cleaning)	32
Figure 3.2	Thesis development timeline (Gantt chart), March–August 2026	46
Figure 4.1	Dataset overview: class balance, category distribution, and content-length distribution by class (cleaned corpus)	48
Figure 4.2	Accuracy and F1-score comparison across all twenty-seven models	67
Figure 4.3	Four-metric grouped comparison — Accuracy, Precision, Recall, and F1-score	69
Figure 4.4	Receiver Operating Characteristic (ROC) curves for all models	64
Figure 4.5	Precision–Recall curves for the top-performing models	65
Figure 4.6	Confusion matrix of the best model (Soft Voting) — raw counts and normalized	62
Figure 4.7	Training and validation curves for deep-learning and transformer models	51
Figure 4.8	Prediction-correlation heat-map of the ensemble's selected base models	53
Figure 4.9	Feature-importance analysis — LightGBM gain-based ranking and Logistic Regression top coefficients	56
Figure 4.10	Error analysis — accuracy by news category and by content length	59
Figure 4.11	SHAP summary plot of the hand-crafted meta-features	57
Figure 4.12	LIME local explanation for an individual test article	58
Figure 4.13	Hyperparameter tuning curve for Logistic Regression (mean 5-fold CV F1-score vs. regularisation strength C).	70
Figure 4.14	t-SNE projection of the 300-dimension SVD document representation for the test partition, coloured by true label.	71
Figure 4.15	Distribution of twelve hand-crafted meta-features by class (Fake vs. Real), test partition.	72
Figure 4.16	Threshold sensitivity curve for the Soft-Voting ensemble: Precision, Recall, and F1-score across the full 0.05–0.95 decision-threshold range.	73
Figure 4.17	Confusion matrices (counts and row-normalised percentages) for the six best-performing models.	74
Figure 4.18	F1-score of all eight ensemble strategies compared against the best individual model (LightGBM, 86.40% F1).	75

Figure 4.19	PCA projection of the 300-dimension document representation: true labels (left) and Soft-Voting correctness (right).	76
Figure 4.20	McNemar's test p-values for pairwise prediction agreement among the top six models (* = significant, $p < 0.05$).	77
Figure 4.21	Learning curve for Logistic Regression: training and 3-fold cross-validation F1-score vs. training-set size.	78
Figure 4.22	Word clouds of the most frequent terms in cleaned article content, Fake (left) and Real (right).	79
Figure 4.23	Five-fold cross-validation F1-score stability for four classical models with reusable feature matrices.	80
Figure 4.24	Distribution of Soft-Voting predicted probability for correct vs. incorrect test-set predictions.	81
Figure 4.25	Per-category Precision, Recall, and F1-score of the Soft-Voting ensemble on the test partition.	82
Figure 4.26	Per-sample inference latency versus F1-score trade-off across all classical and deep-learning models still resident in memory.	83
Figure 4.27	Multi-metric radar comparison of the top five models (Accuracy, Precision, Recall, F1-score, ROC-AUC).	84
Figure 4.28	Zipf's-law check: word-rank versus frequency for the cleaned corpus vocabulary, log–log scale.	85

LIST OF TABLES

Table No.	Title	Page No
Table 2.1	Comparative analysis of reviewed studies on Bengali and low-resource fake news detection	26
Table 3.1	Text-preprocessing operations applied to the Bengali news corpus and their rationale	33
Table 3.2	Categories of hand-crafted meta-features used in the classical feature space	36
Table 3.3	Summary of all twenty-seven models grouped by methodological paradigm	38
Table 3.4	Stratified train / validation / test split summary	44
Table 4.1	Performance comparison of all twenty-seven models on the held-out test set, ranked by F1-score	54
Table 4.2	Confusion matrix of the best model (Soft-Voting ensemble) on the test set	63
Table 4.3	Per-class Precision, Recall, and F1-score of the best model	63
Table 4.4	Test accuracy by article length band	64
Table 6.1	Mapping of research questions (Chapter 1) to key findings (Chapters 4–5)	95

CHAPTER 1
INTRODUCTION

1.1 Background of the Study

Bangladesh has undergone a remarkable transformation in its information ecosystem over the past decade. What was once a media landscape dominated by print newspapers, terrestrial television, and state-run broadcasting has, within a single generation, been reshaped into one of the most actively networked digital societies in South Asia. This shift has fundamentally altered how the Bengali-speaking population consumes, shares, and trusts information. While this transformation has democratized access to knowledge and given ordinary citizens a voice in public discourse, it has simultaneously created an environment in which unverified, misleading, and deliberately fabricated content can spread with alarming speed and, at times, devastating real-world consequences. Understanding the scale and nature of this problem is essential before any technical solution, such as the machine-learning- and deep-learning-based detection framework developed in this thesis, can be meaningfully justified. The following subsections trace this background systematically: from the digital revolution that enabled mass connectivity, to the rise of Bengali content on social platforms, the conceptual definition of fake news, documented real-world incidents in Bangladesh, the consequences borne by Bengali-speaking communities, and finally the resulting need for an automated, language-specific detection system.

1.1.1 The Digital Revolution in Bangladesh

Bangladesh's digital infrastructure has expanded at an extraordinary pace. By mid-2025, the country had reached roughly 188.45 million mobile phone subscriptions across its major operators Grameenphone, Robi, Banglalink, and Teletalk a figure that exceeds the country's entire population and reflects widespread multi-SIM ownership [1]. Over the same period, internet subscribers grew to approximately 133.61 million, comprising both mobile-internet users and fixed broadband/ISP subscribers, pushing overall internet penetration to roughly 44–47 percent of the population. Household-level internet access rose sharply as well, from about 43.6 percent in 2023 to 54.8 percent in 2025, according to Bangladesh Bureau of Statistics figures, while smartphone ownership climbed from 63.3 percent to 72.8 percent in the same window. Nationwide 4G coverage has effectively reached saturation, with 3G coverage close behind, meaning that even in rural and semi-urban areas a smartphone with a data connection is now the primary gateway through which most citizens experience the internet.

This rapid, largely mobile-first digital revolution has outpaced the parallel development of digital and media literacy programmers. A population that gained internet access mainly through affordable smartphones and cheap data packages, rather than through desktop computing and formal computer education, often encountered social media and messaging applications as its first, and for many its only, window into the wider

information environment. Newsrooms, regulatory bodies, and educational institutions have not been able to scale digital-literacy initiatives at the same speed as network expansion. As a direct consequence, a very large number of first-time and relatively inexperienced internet users were, and continue to be, exposed to unmoderated and unverified Bengali-language content without the contextual skills to critically evaluate it. This gap between infrastructural growth and information literacy forms the structural backdrop against which fake news has been able to flourish in Bangladesh.

1.1.2 Rise of Bengali Content on Social Media

Alongside general internet growth, social-media adoption in Bangladesh has become deeply entrenched in everyday life. Facebook remains by far the dominant platform, with tens of millions of active users the majority of the Bangladeshi user base communicating primarily in Bengali, either in the native script or in Romanised ("Banglish") transliteration. Facebook Messenger, YouTube, and, increasingly, short-video platforms have also become significant channels through which Bengali-language content including news, opinion, satire, and outright fabrication circulates. Crucially, the growth of social media has coincided with a proliferation of small, loosely regulated online "news portals" that publish Bengali content with minimal editorial oversight. Many of these portals operate primarily to generate advertising revenue through page views, and headlines are frequently optimised for shareability and emotional reaction rather than factual accuracy.

This environment has blurred the traditional boundaries between professional journalism, opinion commentary, satire, and deliberate misinformation. Satirical Bengali outlets that publish clearly fictional stories for humour are, in practice, often shared without their satirical framing, and readers unfamiliar with the source may take the content at face value. At the other end of the spectrum, outlets that exist purely to fabricate sensational stories mimic the visual style and tone of legitimate news to appear credible. Because Bengali social-media conversation happens largely within closed or semi-closed groups, family and community networks, and Messenger chains, a fabricated story can travel through trusted personal networks and acquire perceived legitimacy purely through repetition and the credibility of the person who shared it, rather than the credibility of the original source. This social dynamic, characteristic of how Bengali-language communities interact online, is a central reason why fake news in Bangladesh cannot simply be addressed with detection tools built for English or other high-resource languages; the linguistic, stylistic, and platform-specific patterns of Bengali misinformation require dedicated study and dedicated modelling.

1.1.3 Definition and Types of Fake News

For the purposes of this research, fake news is defined as news-style content that is deliberately or knowingly fabricated, distorted, or presented out of context, and disseminated under the guise of legitimate journalism with the intent, or at minimum the practical effect, of misleading readers [2], [3]. This definition deliberately excludes honest reporting errors that are promptly corrected, and it distinguishes fake news from

openly labelled satire, though satire that is stripped of its context and shared as if factual can behave functionally like fake news once it enters social circulation.

Fake news is not a single, uniform phenomenon; it manifests in several recognizable forms, all of which are represented, in varying proportions, within the Bengali news corpus used in this study:

- **Fabricated content:** Entirely invented stories with no factual basis, designed to look like genuine news articles, complete with fictitious quotes, dates, and sources.
- **Manipulated or misleading content:** Genuine events or images that are distorted, selectively edited, or captioned out of context to imply something false.
- **Imposter or spoofed-source content:** Content published under a fabricated or hijacked outlet identity that imitates the branding of a trusted news organisation.
- **False context:** Authentic photographs, videos, or facts from one event that are recirculated and falsely attributed to a different, often more recent, incident to provoke a stronger reaction.
- **Sensational or clickbait content:** Headlines exaggerated far beyond what the underlying article supports, exploiting curiosity or outrage to drive engagement.
- **Propaganda and politically or religiously motivated disinformation:** Content crafted specifically to promote a political, communal, or ideological agenda rather than to inform.

In the dataset compiled for this thesis, 11,839 Bengali news articles were collected across twelve content categories National, International, Politics, Sports, Entertainment, Editorial, Miscellaneous, Lifestyle, Education, Crime, Finance, and Technology and labelled as either genuine ("Real") or fabricated ("Fake"), with the two classes nearly balanced at 50.68 percent and 49.32 percent respectively. This near-even distribution reflects a deliberate effort to construct a corpus suitable for supervised classification without the complications introduced by severe class imbalance, and it captures the diversity of fake-news types described above rather than focusing on a single narrow category such as political misinformation alone.

1.1.4 Real-world Examples of Bangla Fake News Incidents

The abstract risks of misinformation become concrete when examined through documented incidents in Bangladesh, several of which stand among the most serious cases of social-media-driven harm recorded anywhere in South Asia.

In September 2012, communal violence erupted in Ramu, Cox's Bazar, after a fabricated Facebook post tagged a local Buddhist resident to an image of a burned copy of the Quran. The post, whose origin was never conclusively traced to the person tagged in it, triggered attacks that destroyed multiple centuries-old Buddhist temples and homes [4]. This incident is widely cited as the first major case in Bangladesh in which a single fabricated social-media post directly precipitated large-scale communal destruction, and it established a troubling template that would recur in subsequent years, including further incidents in Pabna (2013), Nasirnagar (2016), and Rangpur (2017), each following a

broadly similar pattern of an unverified allegation of religious desecration spreading online ahead of any factual verification.

The most severe recurrence of this pattern took place in October 2021. On the morning of 13 October, an unverified claim that a copy of the Quran had been found placed at the feet of a Hindu deity during Durga Puja celebrations in Cumilla began circulating on Facebook, accompanied by photographs and video clips [4]. Mainstream news outlets were slow to publish verified information, leaving social media as the dominant, and largely unchecked, source of information for hours. The rumour spread to Noakhali, Chandpur, and eventually to districts across the country, including the capital Dhaka, over the following days. By the time the unrest subsided, more than one hundred temples and makeshift worship sites had been vandalised, and at least ten people had been killed nationwide, with the toll continuing to rise as the violence spread. A subsequent investigation identified the individual who had placed the religious text at the pandal, but by then the damage, both physical and social, had already been done. The episode also revealed a secondary layer of misinformation: a widely cited social-media account purporting to represent a minority-rights organisation, and used as a source by numerous international media outlets, was later found to be an unauthorised, fabricated account with no genuine connection to the organisation it claimed to represent.

A separate but related category of harm emerged in July 2019, when rumours spread rapidly on Facebook claiming that child abductors were roaming the country to collect heads and blood as a supposed sacrificial requirement for the construction of the Padma Bridge. Within roughly two weeks, at least eight people were beaten to death by vigilante mobs across different districts, none of whom, according to the police investigation that followed, had any connection to child abduction [5]. The most widely reported case was that of a 40-year-old mother of two who was lynched outside a primary school in Dhaka's Badda area after being wrongly suspected of being a kidnapper while she was there to enquire about her children's school admission. Human-rights monitors recorded dozens of related lynching incidents nationwide within the first half of that year alone, underscoring how quickly an online rumour, entirely disconnected from any verified event, could translate into fatal offline violence.

These incidents, spanning religious, communal, and criminal misinformation, demonstrate that Bangladesh's fake-news problem is neither hypothetical nor confined to minor reputational harm; it has repeatedly resulted in loss of life, destruction of property, and lasting social division, which is precisely the class of real-world consequence that motivates the automated detection approach developed in this research.

1.1.5 Consequences of Misinformation in Bengali-Speaking Communities

The consequences of unchecked misinformation in the Bengali-speaking context extend across several interconnected dimensions:

- **Loss of Life and Physical Violence:** Incidents of mob lynching and killing in Ramu, Cumilla, and those surrounding the Padma Bridge rumours demonstrate that, over the

past decade, fabricated or unverified claims have repeatedly triggered deadly acts of mob violence.

- **Communal and Religious Polarisation:** Recurrent attacks targeting religious minorities have deepened mistrust among different communities. Researchers have documented several cases indicating that these incidents were deliberately orchestrated to intimidate or displace minority populations.
- **Erosion of Trust in Media and Institutions:** When rumours spread faster than verified information as witnessed during the unrest in Cumilla, where unverified social-media content circulated far more rapidly than reports from mainstream media public trust in both social-media platforms and traditional journalism is undermined. As a result, people become more susceptible to believing future rumours while simultaneously growing sceptical of accurate information and official corrections.
- **Economic Losses and Property Damage:** Unrest fuelled by misinformation has resulted in the vandalism and destruction of temples, homes, and business establishments, causing significant financial losses for already vulnerable communities while diverting the resources of law-enforcement agencies from crime prevention to post-incident investigation and response.
- **Psychological Trauma:** The families of victims killed in mob lynchings, survivors of such attacks, and residents of affected communities often experience long-term psychological trauma that is rarely reflected in casualty statistics alone.
- **Use as a Political and Electoral Tool:** Analysts have observed that the spread of communal misinformation in Bangladesh frequently coincides with politically sensitive periods, suggesting that fabricated content is not always the result of spontaneous rumour but is, at times, strategically disseminated to incite political or communal tension.

Overall, it is evident that, in the context of the Bengali language, fake news is not a marginal or insignificant nuisance; rather, it represents a serious threat to public safety and social cohesion. Its widespread prevalence and the significant human costs associated with its consequences highlight the urgent need for sustained academic research and the development of effective technological solutions to address this growing challenge.

1.1.6 Need for Automated Fake News Detection in Bangla

Given the rapid spread of misinformation on Bengali social media and the severe consequences it can produce, relying solely on manual fact-checking is neither practical nor sufficient. Regardless of the dedication and expertise of fact-checking organisations and verification teams within news media, the verification process often requires several hours to complete. In contrast, a fabricated post can be shared through Facebook groups and Messenger chains, reaching millions of users within minutes. The Cumilla incident described above illustrates this disparity clearly: unverified claims had already gone viral before verified reports were published, and it was during this critical time gap that unrest and tensions began to escalate. Addressing this challenge requires an automated and scalable system capable of flagging suspicious Bengali news content in near real time.

Such a system would enable social-media platforms, fact-checking organisations, and news consumers to be alerted before misinformation causes widespread harm.

However, developing such a system for the Bengali language is considerably more challenging than for high-resource languages such as English. Bengali remains relatively under-resourced in terms of natural-language-processing (NLP) tooling. Robust stemming and lemmatisation techniques are limited, spelling conventions and word boundaries often vary across informal social-media posts and online news articles, and textual content frequently combines Bengali script with Romanised transliteration, English loanwords, emojis, and platform-specific elements such as hashtags, URLs, and advertising fragments. Consequently, an effective fake-news detection system for Bengali must be tailored to the language's unique linguistic characteristics rather than relying on methods directly adapted from English-language fake-news detection studies.

These challenges have significantly influenced the methodological design of this research. Accordingly, this thesis adopts Unicode normalisation during text preprocessing, deliberately avoids unreliable Bengali stemming, employs character-level n-gram representations to capture spelling variation and rich morphological patterns, and fine-tunes both a Bengali-specific and a multilingual transformer model (BanglaBERT and XLM-RoBERTa respectively). These approaches are evaluated alongside classical machine learning and deep-learning models to develop a robust and effective framework for Bengali fake-news detection.

It is equally important to avoid methodological “shortcuts” that may appear effective on paper but fail in real-world deployment. One such flawed approach is to classify news articles as fake based on the names of unreliable news outlets or technical artifacts inadvertently retained during web scraping, rather than learning the genuine linguistic and stylistic characteristics of deceptive content. If a detection system merely memorises the source website instead of learning the underlying patterns of fake Bengali news, it will generalise poorly to previously unseen sources and will provide little practical protection against newly emerging misinformation. Section 1.2 and Chapter 3 return to this issue in detail, since the identification and removal of exactly such shortcuts forms one of the central methodological contributions of this thesis.

Considering Bangladesh's rapid digital transformation, the dominance of the Bengali language across social-media platforms, and the tangible sometimes fatal consequences of uncontrolled misinformation, this thesis proposes, implements, and rigorously evaluates a comprehensive automated framework for Bengali fake-news detection. The proposed framework integrates carefully engineered linguistic and stylistic features, classical machine-learning classifiers, deep-learning architectures, transformer-based language models, and ensemble-learning techniques, all trained and evaluated on a carefully curated and deduplicated dataset of Bengali news articles. The primary objective of this research is to develop a transparent, reproducible, and practically deployable fake-news detection framework that can contribute to mitigating the spread of misinformation in the Bengali language.

1.2 Problem Statement

The overarching problem addressed by this thesis can be stated concisely as follows: given a Bengali news article consisting of a category label, a headline, and a body of content, design and rigorously validate a system that automatically classifies the article as real or fake, in a manner that generalises to genuinely unseen content rather than exploiting artefacts of how the training data happened to be collected. The following subsections unpack the specific dimensions of this problem that motivate the objectives set out in Section 1.3.

1.2.1 Ineffectiveness of English-based Detection Systems for Bangla

The majority of currently available fake-news detection frameworks have primarily been developed and optimised for the English language. These systems take advantage of decades of linguistic research, large-scale annotated datasets, and advanced NLP resources. Although these frameworks have demonstrated remarkable effectiveness on English datasets often achieving accuracy levels above 95 percent on benchmark corpora such as LIAR and FakeNewsNet [6], [7] they are built upon assumptions and linguistic resources that are not directly applicable to Bengali. Pre-trained language models such as BERT, RoBERTa, and GPT have achieved remarkable success in understanding English text; however, they require extensive fine-tuning on language-specific data and cannot be directly applied to Bengali without significant adaptation.

Attempts to apply English-based fake-news detection systems to Bengali content through simple translation or cross-lingual transfer have proven inadequate. Machine translation often introduces additional noise and removes important linguistic nuances that are essential for identifying deceptive patterns. Moreover, cultural and contextual differences between Bengali-speaking and English-speaking communities influence the way misinformation is expressed: linguistic indicators of fake news such as sensational language, emotional manipulation, exaggerated claims, and logical fallacies may appear differently depending on the language and socio-cultural context.

The fundamental dependence of English-based systems on English-centric linguistic resources such as syntactic parsers, named-entity-recognition tools, and sentiment lexicons makes them largely ineffective when directly applied to Bengali text. Even multilingual models such as XLM-RoBERTa and mBERT, which provide some degree of cross-lingual transfer capability, often demonstrate considerably lower performance for Bengali compared with English. In the experiments reported in this thesis (Chapter 4), it was observed that XLM-RoBERTa achieved a test F1-score of 83.35 percent, appreciably lower than several carefully engineered classical machine-learning models trained directly on properly preprocessed Bengali text. This performance gap underlines the need for detection systems that are specifically designed around the linguistic, contextual, and cultural characteristics of Bengali content, rather than adapted post hoc from English-centric pipelines.

1.2.2 Linguistic Complexities of Bengali Language

The Bengali language presents several unique linguistic challenges that make automatic fake-news detection a genuinely difficult task. One major obstacle is the rich morphological structure of Bengali. Whereas English relies primarily on word order and sentence structure to express grammatical relationships, Bengali extensively uses inflection to represent tense, aspect, mood, person, and honorific register. A single root word can appear in numerous surface forms, which substantially increases vocabulary size and creates serious data-sparsity problems for machine-learning models. This morphological complexity makes feature extraction particularly challenging: bag-of-words and TF-IDF representations become highly dimensional and sparse, while the absence of reliable Bengali stemmers and lemmatisers makes effective vocabulary reduction difficult.

The Bengali writing system also introduces additional computational challenges. The script, derived from the Brahmi family, contains complex conjunct-character combinations (যুক্তাক্ষর), requiring careful Unicode normalisation to ensure consistent processing. Variations in character representation, inconsistent use of the Zero-Width Joiner (ZWJ) and Zero-Width Non-Joiner (ZWNJ), and the presence of diacritical marks can produce multiple valid encodings for what is visually identical text. Without proper normalisation, these variations can create inconsistencies between training and testing data, ultimately reducing model performance and generalisation ability a concern this thesis addresses directly through NFC (Normalisation Form Canonical Composition) normalisation in the preprocessing pipeline (Section 3.4.1).

Furthermore, the increasing use of "Benglish" the combination of Bengali and English within the same text or sentence adds another layer of complexity. Benglish appears in several forms, including the use of Latin script for Bengali words (Romanised Bengali), the inclusion of English words within Bengali sentences, and the mixing of both languages in a single expression. This phenomenon is particularly common in social-media content, where a significant portion of misinformation spreads. Models trained exclusively on standard Bengali text often struggle to process and classify such mixed-language inputs effectively.

The presence of multiple regional dialects, differences between formal journalistic language and informal conversational language, and the lack of standardised spelling conventions further intensify these challenges. Fake-news content often exploits these linguistic variations by using emotional, sensational, and exaggerated language patterns that differ from conventional journalistic writing style. Moreover, the limited availability of comprehensive Bengali linguistic resources such as large-scale corpora, part-of-speech taggers, named-entity-recognition systems, and sentiment lexicons makes addressing these challenges even more difficult.

Therefore, developing an effective fake-news detection system for the Bengali language requires language-specific preprocessing strategies, robust feature-representation

techniques, and models capable of accommodating the diverse linguistic characteristics of Bengali content, rather than a direct transplant of English-oriented NLP pipelines.

1.2.3 Scarcity of Labeled Bangla Fake News Datasets

The development of effective supervised machine-learning models for fake-news detection depends heavily on the availability of high-quality, well-annotated datasets. In English fake-news research, several large-scale benchmark datasets are available, such as LIAR (more than 12,000 labelled statements) [6] and FakeNewsNet (thousands of articles with accompanying social context) [7]. The scenario is markedly different for Bengali. The dataset used in this research, containing 11,839 news articles labelled as either fake or authentic, represents one of the larger available collections for Bengali fake-news detection, comparable in scale to the BanFakeNews corpus [8], yet it remains considerably smaller than equivalent English-language resources.

The scarcity of labelled Bengali fake-news datasets can be attributed to several factors. First, the annotation process is highly labour-intensive and requires individuals with expertise in both linguistic analysis and journalism. Fake-news detection is inherently a somewhat subjective task, because the distinction between misinformation, satire, personal opinion, and authentic reporting is often unclear; different annotators may interpret the same content differently, resulting in label noise and inconsistent ground truth. Analysis of the dataset used in this research (Section 3.3) revealed that 155 content groups contained conflicting labels, affecting 310 individual records approximately 2.6 percent of the raw corpus and such levels of label noise can meaningfully constrain the achievable performance of supervised learning models.

Second, the cost of expert-level annotation is often beyond the resources available to individual research teams, particularly when a dataset must cover diverse domains such as politics, economics, healthcare, sport, and entertainment, each requiring some degree of domain-specific judgement to label accurately. Third, the constantly evolving nature of fake news makes existing datasets become outdated quickly: new forms of misinformation, changing linguistic patterns, and shifting political and social contexts continuously emerge, requiring ongoing data collection and re-annotation.

The limited size of available datasets has a direct impact on model development and performance. Deep-learning approaches, and transformer-based models such as BanglaBERT in particular, generally require large volumes of training data to reach their optimal performance. In this research, BanglaBERT achieved a test F1-score of only 78.32 percent (76.26 percent accuracy), noticeably lower than several relatively simple ensemble-based approaches (Chapter 4). This performance gap indicates that, although transformer-based models are capable of learning complex linguistic structures and contextual patterns, their full potential cannot be realised when the volume of task-specific training data is limited. As a result, researchers working with Bengali fake-news data are often required to rely on less data-hungry classical machine-learning approaches, or to compensate through resource-intensive ensemble strategies, as is done extensively in this thesis. This highlights the continuing need for larger, more diverse, and accurately

annotated Bengali fake-news datasets to support the next generation of high-performing detection systems.

1.2.4 Absence of Real-time Detection Tools for Bangla Content

Despite the increasing prevalence and severity of misinformation in the Bengali language, there remains a significant lack of accessible, real-time detection tools for Bengali-speaking communities, journalists, and fact-checkers. While numerous automated tools are available for verifying English-language content including browser extensions, social-media API integrations, and dedicated fact-checking platforms the development of comparable solutions for Bengali content has received markedly less attention from both commercial and open-source developers.

This gap exists at multiple levels. Social-media platforms such as Facebook, YouTube, and WhatsApp, which serve as the principal channels for the spread of fake news in Bangladesh, generally rely on content-moderation systems primarily optimised for English-language content. Although these platforms have introduced global initiatives to combat misinformation, their Bengali-specific detection capabilities remain limited owing to the linguistic complexity of Bengali and comparatively lower investment in non-English-language markets. Journalists and professional fact-checkers working in Bangladesh also face a shortage of specialised verification tools and largely depend on manual verification processes; while valuable, these approaches are time-consuming, labour-intensive, and insufficient relative to the volume of content requiring verification, particularly during critical periods such as elections, natural disasters, or public-health emergencies, when the speed of misinformation propagation often exceeds the capacity of manual fact-checking.

Furthermore, even where effective Bengali fake-news-detection models exist, a substantial gap remains between research prototypes and practical deployment. Many advanced models require significant computational resources, making their integration into lightweight applications, browser extensions, or mobile platforms challenging a particularly important consideration in Bangladesh, where mobile-based access is the primary means by which most citizens reach the internet. This research attempts to address this limitation by explicitly comparing model families not only on predictive accuracy but also on computational footprint, and by identifying gradient-boosting models such as LightGBM, XGBoost, and CatBoost which combine strong accuracy with modest inference cost as practical candidates for real-world, resource-constrained deployment (Section 5.8.5).

1.2.5 Social and Political Impact of Unchecked Fake News in Bangladesh

The impact of unchecked fake news in Bangladesh extends far beyond isolated incidents of misinformation; it has profound consequences for social cohesion, political stability, and public health. The rapid spread of false information through social-media platforms has created an environment in which fabricated narratives can easily gain public acceptance and frequently lead to harmful, sometimes destructive, outcomes.

During national election cycles, fabricated news regarding political candidates, allegations of electoral irregularities, and misleading information about political parties has been widely circulated through social-media platforms. Such false narratives contribute to increasing public distrust in the electoral process, intensify political tension, and, in some cases, encourage confrontation between political groups. The absence of reliable and immediate fact-checking mechanisms allows such narratives to spread widely, negatively affecting the transparency and credibility of democratic processes. Public health has been similarly affected: during the COVID-19 pandemic, fake news concerning medical treatments, vaccine side effects, and the origin and severity of the virus spread extensively across Bengali digital platforms, contributing to vaccine hesitancy in a meaningful share of the population.

Communal violence, as documented in Section 1.1.4, represents perhaps the most devastating consequence of unchecked fake news, while the economic impact of misinformation through false claims about economic indicators, corporate misconduct, or government policy can contribute to market instability and damage investor confidence. Taken together, these dimensions demonstrate that the uncontrolled spread of fake news in Bangladesh constitutes a multidimensional challenge affecting information reliability, social harmony, democratic process, public safety, and economic stability simultaneously. Developing an accurate and efficient Bangla fake-news detection system is therefore not merely an academic exercise but a practically urgent contribution towards a safer digital information environment.

1.3 Objectives of the Study

This research is driven by the need to develop an effective automated system for detecting fake news in Bengali. To address the challenges outlined in the problem statement, the study establishes five interconnected research objectives that together form a comprehensive framework for Bangla fake-news detection. Each objective addresses a specific facet of the problem and contributes to the overarching goal of a robust, generalisable detection system.

1.3.1 Objective 1: Dataset Construction

The first and most fundamental objective is to construct, audit, and clean a well-annotated corpus of Bengali news articles suitable for supervised machine learning. Given the scarcity of labelled Bangla fake-news datasets identified in Section 1.2.3, this objective addresses a critical gap in the research landscape and comprises: (i) consolidating a corpus of 11,839 Bengali news articles spanning twelve topical categories; (ii) systematically auditing the corpus for missing values, duplicate content, and conflicting labels; and (iii) implementing a rigorous deduplication and conflict-resolution pipeline, applied strictly before any train/test partitioning, so that no article can appear in both the training and test sets and inflate reported performance through data leakage.

1.3.2 Objective 2: Bangla Text Preprocessing Pipeline

The second objective is to design and implement an effective text-preprocessing pipeline specifically tailored to the linguistic characteristics of Bengali, recognising that raw Bangla text contains multiple forms of noise that must be addressed before meaningful feature extraction can occur. This objective encompasses Unicode NFC normalisation; systematic removal of HTML, URL, JavaScript, advertisement, and web-scraping artefacts; removal of publication-outlet name-tokens that act as unintended shortcuts rather than genuine linguistic signals; principled handling of emojis, hashtags, and elongated expressions; and a deliberate decision to preserve full word forms rather than apply unreliable Bengali stemming or lemmatisation, given the current absence of a broadly validated, production-grade tool for this purpose.

1.3.3 Objective 3: Feature Extraction from Bangla Text

The third objective is to design a comprehensive feature-extraction framework that captures both traditional statistical signal and modern contextual representation from preprocessed Bangla text, on the premise that no single feature family is sufficient on its own. This objective spans: word-level TF-IDF vectorisation (1–3 grams); character-level TF-IDF vectorisation (2–5 grams, `char_wb` analyser) to capture sub-word morphology and spelling variation; a set of 31 hand-crafted linguistic, stylometric, and journalistic-rigour meta-features; Word2Vec-style distributed word embeddings; and contextual embeddings obtained from a pre-trained Bengali transformer (`sagorsarker/bangla-bert-base`). A further sub-goal is the principled integration and scaling of these heterogeneous feature families into unified matrices suitable for downstream classifiers.

1.3.4 Objective 4: Model Development and Training

The fourth objective is to develop, train, and fairly compare a deliberately broad set of classification approaches, on the premise that different modelling paradigms capture different aspects of the detection task and that this diversity can subsequently be exploited through ensembling. This objective covers twelve classical machine-learning algorithms (Logistic Regression, Linear and RBF Support Vector Machines, Naive Bayes, Random Forest, Extra Trees, AdaBoost, Gradient Boosting, XGBoost, LightGBM, CatBoost, and K-Nearest Neighbors), five deep-learning architectures (TextCNN, BiLSTM, GRU, CNN-BiLSTM, and BiLSTM with Attention), two fine-tuned transformer models (BanglaBERT and XLM-RoBERTa), and eight ensemble-combination strategies (soft voting, hard voting, weighted voting, logistic-regression blending, k-fold stacking, rank averaging, Caruana greedy selection, and ridge blending) twenty-seven models in total, all trained under an identical experimental protocol to ensure a fair basis for comparison.

1.3.5 Objective 5: Performance Evaluation and Comparative Analysis

The fifth and final objective is to evaluate all developed models comprehensively and to identify the most effective approach for Bangla fake-news detection while understanding the relative strengths and weaknesses of each methodology. This objective includes: evaluating models with Accuracy, Precision, Recall, F1-score, and ROC-AUC;

conducting systematic comparative analysis across classical machine learning, deep learning, transformer, and ensemble paradigms; optimising decision thresholds on the validation partition; performing detailed error analysis to characterise which content is most frequently misclassified and why; applying explainable-AI techniques (SHAP, LIME, and feature-importance analysis) to verify that models rely on genuine linguistic signal rather than collection artefacts; and, ultimately, establishing an honest, leakage-aware performance benchmark for this dataset rather than reporting an inflated accuracy figure obtained through data-leakage shortcuts.

1.4 Research Questions

Building directly on the objectives above, this thesis is organised around five specific research questions, each of which is revisited explicitly in the discussion of results in Section 4.11 once the corresponding experimental evidence has been presented.

1.4.1 Effective Preprocessing for Bangla Text

RQ1: Which combination of preprocessing operations Unicode normalisation, noise and artefact removal, outlet-marker stripping, punctuation handling, and the deliberate avoidance of stemming/lemmatisation most effectively prepares raw, web-scraped Bengali news text for reliable, leakage-free classification, and what is the measurable consequence of omitting any one of these steps?

1.4.2 Optimal Feature Extraction Technique

RQ2: Among word-level TF-IDF, character-level TF-IDF, hand-crafted stylometric meta-features, Word2Vec embeddings, and BanglaBERT contextual embeddings, which representation, or combination of representations, contributes most strongly to distinguishing genuine from fabricated Bengali news content?

1.4.3 Deep Learning vs Traditional ML Performance

RQ3: Under a dataset of approximately ten thousand cleaned articles, do deep-learning architectures (TextCNN, BiLSTM, GRU, CNN-BiLSTM, BiLSTM-Attention) outperform classical machine-learning models trained on TF-IDF and meta-features, or does the comparatively modest size of the corpus favour simpler, less data-hungry approaches?

1.4.4 Effectiveness of Bangla-BERT

RQ4: How does fine-tuning a Bengali-specific pre-trained transformer (BanglaBERT) compare with fine-tuning a general-purpose multilingual transformer (XLM-RoBERTa) and with the best classical and ensemble models, and what does this comparison reveal about the data requirements of transformer-based approaches in a low-resource setting?

1.4.5 Linguistic Cues for Fake News Identification

RQ5: Which linguistic, stylistic, and structural cues such as lexical diversity, character entropy, headline–content overlap, sensational punctuation, and journalistic-rigour

markers (quotation verbs, named designations, dates, and organisation references) are most predictive of the real/fake distinction, and do the models' learned decision boundaries rely on these genuine cues rather than on incidental artefacts of how the dataset was collected?

1.5 Significance of the Research

The significance of this research extends across academic, social, technological, economic, and civic-institutional dimensions, each of which is discussed below.

1.5.1 Academic Contribution to Bengali NLP

Bengali remains a comparatively low-resource language within the global NLP research landscape, and empirical studies that rigorously compare classical machine learning, deep learning, and transformer-based approaches under a single, leakage-free protocol are scarce. By systematically documenting a full experimental pipeline including a transparent data-quality audit, a reusable preprocessing and feature-engineering methodology, and a twenty-seven-model comparison this thesis contributes a reproducible reference point that future Bengali NLP research, both on fake news and on adjacent text-classification tasks, can build upon.

1.5.2 Social Impact on Bangladeshi Society

As documented extensively in Section 1.1, misinformation in the Bengali-speaking context has directly contributed to communal violence, mob killings, and public-health harm. A reliable automated detection system, even if deployed initially as a decision-support tool rather than an autonomous filter, can shorten the critical window between the emergence of a fabricated claim and its identification, thereby contributing to the mitigation of exactly the kind of real-world harm catalogued in this chapter.

1.5.3 Technological Contribution for News Agencies

News agencies, digital newsrooms, and content-moderation teams operating in Bengali currently lack mature, language-specific automated verification tooling. The trained models, feature pipeline, and explainability analysis developed in this thesis provide a concrete technical foundation including specific guidance on which model families offer the best accuracy-to-computational-cost trade-off (Section 5.8) that can inform the design of in-house verification assistants for Bengali-language publishers.

1.5.4 Economic and Commercial Potential

Beyond direct social benefit, an efficient Bengali fake-news classifier has commercial value for social-media platforms, digital-advertising networks seeking to avoid placing advertisements alongside disinformation, and media-monitoring firms that serve corporate and governmental clients. Because the strongest individual models identified in this study (LightGBM, XGBoost, CatBoost) are computationally lightweight relative to transformer-based alternatives, the resulting pipeline is realistically deployable at the scale and cost profile required by such commercial applications.

1.5.5 Support for Fact-Checking Organizations

Bangladeshi fact-checking organisations currently rely almost entirely on manual verification workflows. An automated pre-filtering system, of the kind developed in this thesis, can triage incoming content by estimated likelihood of being fabricated, allowing scarce human fact-checking capacity to be directed toward the articles most likely to require expert attention improving both the speed and the coverage of the verification pipeline as a whole.

1.6 Scope and Limitations

1.6.1 Scope of the Research

The scope of this research is deliberately bounded to ensure that the findings reported in later chapters rest on a well-defined and internally consistent experimental protocol.

1.6.1.1 Text-based Bangla Content Only

The research is confined to the textual content of news articles headlines and body text and does not process accompanying images, videos, or audio, even where such media are known to carry independent misinformation signal.

1.6.1.2 Supervised Learning Approach

All twenty-seven models are trained under a supervised learning paradigm using binary ground-truth labels (real/fake); unsupervised and semi-supervised detection strategies fall outside the scope of this study.

1.6.1.3 News Articles and Social Media Posts

The dataset consists of structured news articles gathered from Bengali news portals rather than raw, unstructured social-media posts; consequently, the conclusions drawn most directly concern news-style content rather than informal social-media chatter.

1.6.1.4 Offline Classification System

The system developed and evaluated in this thesis is an offline batch-classification pipeline. It establishes the modelling foundation for, but does not itself implement, a production real-time monitoring or streaming-detection service.

1.6.2 Limitations of the Research

The following limitations are acknowledged and discussed in greater depth in Chapter 5.

1.6.2.1 Limited Dataset Size

At approximately ten thousand clean articles, the corpus is modest by the standards of modern deep-learning and transformer research, constraining the extent to which data-hungry architectures can realise their full potential (Section 1.2.3, Section 5.2.1).

1.6.2.2 Inability to Detect Image/Video Fake News

Because the pipeline is text-only, multimodal misinformation manipulated images, out-of-context video, or deepfake content accompanying an otherwise plausible textual narrative cannot be detected by the system as developed.

1.6.2.3 Challenges with Bengali Dialects

The dataset is dominated by formal, standard Bengali as used in published news writing; regional dialectal variation and highly informal conversational registers are under-represented and may reduce model reliability if applied to such content.

1.6.2.4 No Real-time Processing Capability

Consistent with Section 1.6.1.4, the models are validated in an offline evaluation setting and have not been benchmarked for streaming throughput, latency, or integration with a live content-ingestion pipeline.

1.6.2.5 Poor Performance on "Benglish" (Mixed Language)

Romanised Bengali and English–Bengali code-mixed text, common on social media but rare in the formally published news corpus used here, is expected to be handled poorly by the trained models, since the vocabulary and character-level representations were learned almost exclusively from standard Bengali script.

1.7 Organization of the Thesis

The remainder of this thesis is organised into six further chapters.

- **Chapter 2 Literature Review** reviews the existing literature on fake-news detection from a global perspective and specifically for Bengali, culminating in a structured comparison table and an explicit statement of the research gap this thesis addresses.
- **Chapter 3 Research Methodology** describes the complete research methodology: the dataset and its quality audit, the Bangla-specific preprocessing pipeline, the feature-extraction framework, every model family and ensemble strategy used, and the experimental setup and evaluation metrics.
- **Chapter 4 Results and Discussion** presents the exploratory data analysis and the full experimental results for all twenty-seven models, together with visualisations, feature-importance and error analysis, and a discussion that maps findings back to the five research questions of Section 1.4.
- **Chapter 5 Limitations and Ethical Considerations** critically evaluates model generalisation, the overfitting/underfitting behaviour observed across paradigms, and the practical and ethical considerations privacy, fairness, transparency, and responsible deployment that must accompany any real-world use of the system.
- **Chapter 6 Conclusion and Future Work** synthesises the key findings, restates the contributions of the study, and outlines concrete directions for future work, including dataset expansion, multimodal detection, and real-time deployment.

The thesis concludes with a full IEEE-style reference list and an appendix containing representative dataset samples, key code excerpts from the experimental pipeline, and supplementary result tables that support, without cluttering, the discussion in the main chapters.

CHAPTER 2
LITERATURE REVIEW

2.1 Introduction

Research on automated fake-news detection has grown into a substantial field over the past decade, driven overwhelmingly by work on English-language datasets and models. For Bengali, by contrast, the body of published literature remains comparatively small, fragmented across a handful of dataset papers and isolated modelling studies. This chapter situates the present thesis within that literature. Section 2.2 surveys the global evolution of fake-news detection, from rule-based heuristics through to modern transformer-based classifiers. Section 2.3 considers the specific difficulties that arise when this task is transplanted to low-resource languages in general. Section 2.4 reviews the small but growing body of work specifically targeting Bengali fake news, grouped by whether the primary contribution is a dataset, a feature-engineering scheme, or a deep-learning/transformer architecture. Sections 2.5 through 2.7 review, respectively, the tooling and linguistic challenges specific to Bengali NLP, the feature representation techniques applicable to text classification in general, and ensemble learning as a family of techniques for combining model predictions. Section 2.8 presents a structured, criteria-based comparison of the studies most directly relevant to this thesis, and Section 2.9 uses that comparison to state explicitly the research gap this work addresses.

2.2 Fake News Detection: A Global Perspective

The methodological history of automated fake-news detection can be read as a progression through four broadly overlapping generations of technique, each building on the representational capacity of the last.

2.2.1 Traditional Approaches (Rule-based and Linguistic)

Early work on deception and misinformation detection relied on explicit linguistic and rhetorical rules: hand-crafted lists of sensationalist vocabulary, counts of exclamation marks and superlatives, readability scores, and simple source-credibility heuristics such as domain blacklists. These approaches are transparent and cheap to compute, but brittle they generalise poorly to novel phrasing and are trivially evaded once the underlying rule set becomes known. Nonetheless, several of the linguistic intuitions from this era particularly the idea that sensationalism, exaggerated punctuation, and a lack of attributable sourcing are weak but genuine correlates of fabricated content persist in the hand-crafted stylometric and journalistic-rigour features used by this thesis (Section 3.5.3).

This generation of work also drew on forensic-linguistics research (Undeutsch's hypothesis) that fabricated narratives differ measurably from genuine ones fewer self-referencing pronouns, fewer sensory details, more uniformly exaggerated affect. Several

of these early intuitions persist, largely unchanged, in the stylometric feature set used in this thesis (Section 3.5.3).

2.2.2 Machine Learning-based Detection Approaches

The dominant paradigm through the 2010s combined statistical text representations, chiefly bag-of-words and TF-IDF vectors [18], with supervised classifiers such as Naive Bayes, Logistic Regression, and Support Vector Machines [26]. Tree-based ensembles Random Forest [19], AdaBoost [20], Gradient Boosting, and later the highly optimised gradient-boosting libraries XGBoost [21], LightGBM [22], and CatBoost [23] subsequently became the strongest-performing family of classical models for high-dimensional, sparse text-classification tasks, a finding this thesis independently reproduces in Chapter 4, where gradient-boosting models are the strongest individual classifiers on the Bengali corpus.

A key advantage of this paradigm is computational efficiency: sparse TF-IDF matrices train in seconds to minutes on CPU hardware, versus GPU-hours for deep models. This advantage is not merely historical Section 4.3 and Figure 4.26 show the classical paradigm remains both the most efficient and the most accurate individual-model family on this corpus.

2.2.3 Deep Learning-based Detection Approaches

From roughly 2014 onward, neural architectures able to learn distributed representations directly from raw or lightly processed text began to outperform purely statistical pipelines on many benchmarks. Convolutional architectures such as TextCNN [29] proved effective at capturing local, n-gram-like phrase patterns, while recurrent architectures LSTM [27] and GRU [28], particularly in their bidirectional form captured longer-range sequential dependency. Attention mechanisms [30] further allowed such networks to weight the most informative tokens in a sequence, improving both accuracy and interpretability; the BiLSTM-with-Attention architecture used in this thesis (Section 3.7.4) draws directly on this line of work.

A recurring finding, directly relevant to Section 4.4, is that this representational advantage is conditional on sufficient task-specific training data: with only a few thousand examples, a network trained from scratch must learn embeddings and classification jointly, whereas a classical model benefits from structure already supplied by feature engineering the central empirical question behind Research Question 3 (Section 1.4.3).

2.2.4 Transformer and Pretrained Language Model Approaches

The introduction of the Transformer architecture [15] and subsequent pretrained language models such as BERT [14] established a new state of the art across most NLP tasks, including fake-news detection, by learning deep bidirectional contextual representations from very large unlabelled corpora and then fine-tuning them on comparatively small task-specific datasets. Multilingual variants, most notably XLM-RoBERTa [13], extend this capability across roughly one hundred languages, including Bengali, using a single

shared model. However, as this thesis demonstrates empirically (Sections 4.4 and 5.2.4), the benefit of transformer fine-tuning is strongly conditional on the availability of sufficient task-specific training data a condition that is not comfortably met by current Bengali fake-news corpora, including the one constructed for this study.

Pretrained language models are exposed to orders of magnitude more raw text during pretraining than any downstream task supplies as labelled fine-tuning data, which is what normally allows fine-tuning to succeed on small task-specific corpora. Whether this advantage is sufficient here is exactly the empirical question the BanglaBERT/XLM-RoBERTa comparison in Sections 4.5 and 4.11.4 addresses.

2.3 Fake News Detection in Low-Resource Languages

Low-resource languages those lacking the large annotated corpora, mature NLP tooling, and pretrained model ecosystems available for English pose a distinct set of challenges for fake-news detection research. Three recurring issues are especially relevant to this thesis. First, dataset scarcity forces researchers to work with corpora one to three orders of magnitude smaller than mainstream English benchmarks such as LIAR [6] or FakeNewsNet [7], which in turn constrains the choice of viable modelling approaches and disadvantages data-hungry deep architectures. Second, publicly released low-resource corpora are frequently assembled by web-scraping a small number of source outlets for each class, which risks the model learning source-identity or collection-pipeline artefacts rather than genuine content signal precisely the phenomenon investigated in depth in Section 3.3.5 of this thesis. Third, general-purpose multilingual models, while convenient, typically under-perform relative to models either trained from scratch on the target language or fine-tuned from a language-specific pretrained checkpoint, because the multilingual model's shared parameter budget is spread thinly across dozens of typologically diverse languages. Bengali fake-news research inherits all three of these constraints, and this thesis's methodological choices a leakage-aware data-quality audit, a Bengali-specific preprocessing pipeline, and a direct comparison between a Bengali-specific transformer (BanglaBERT) and a multilingual one (XLM-RoBERTa) are designed specifically to confront them rather than to assume they do not apply.

2.4 Existing Research on Bengali Fake News Detection

Published research specifically targeting Bengali fake-news detection can be usefully grouped into three, not mutually exclusive, categories: studies whose primary contribution is a labelled dataset, studies centred on feature-engineering and classical machine learning, and studies centred on deep-learning or transformer architectures. The two studies analysed in the greatest depth for this thesis (and summarised in Table 2.1) fall respectively into the third and second categories.

2.4.1 Dataset-oriented Studies

The most widely used public resource in this space is BanFakeNews [8], a corpus of approximately fifty thousand Bengali news articles combining authentic articles collected

from established, verified news portals with fake, satirical, and clickbait articles gathered through a combination of web sources and crowdsourcing. BanFakeNews established several methodological conventions that later work, including both studies reviewed in Section 2.4.2–2.4.3 and, indirectly, this thesis, continues to build on: a multi-category labelling scheme distinguishing outright fabrication from satire and clickbait, provision of rich per-article metadata (publication domain, date, category, and inferred "relation" to any real story being distorted), and an initial set of baseline classical and neural benchmarks. The dataset's principal limitation, from the perspective of this thesis, is a pronounced class imbalance the original release contains on the order of forty-eight thousand authentic articles alongside roughly thirteen hundred fake articles together with the fact that authentic and fake articles were drawn from different, largely non-overlapping sets of source portals, creating exactly the kind of source-identity shortcut risk that this thesis's leakage audit (Section 3.3.5) is designed to detect and remove. Partly for this reason, this thesis constructs and audits an independent, near-balanced 11,839-article corpus rather than reusing BanFakeNews directly, while continuing to treat it as the primary comparison point for the wider literature.

2.4.2 Feature Engineering-based Studies

A representative feature-engineering-centred study [10] proposes a two-layer stacked machine-learning system built around TF-IDF vectorisation. Five classical classifiers SVM, Random Forest, Logistic Regression, Gradient Boosting, and XGBoost serve as layer-one base learners, and a Logistic Regression meta-model, trained on the base learners' predictions, produces the final decision. The study additionally explores mBERT and Word2Vec as alternative vectorisation strategies and evaluates the pipeline across three datasets: one English corpus and two Bangla corpora (including BanFakeNews). It reports 99.6% accuracy and a 99.8% F1-score on the English dataset, an improvement from 74.8% to 85.2% accuracy on a smaller, imbalanced Bangla dataset after applying SMOTE oversampling, and 98.4% accuracy on BanFakeNews. This work is methodologically the closest antecedent to the ensemble strategy adopted in this thesis (Section 3.9): both combine multiple base learners through a trained meta-model. It differs, however, in three important respects that motivate parts of this thesis's design: it does not report a duplicate-record or label-conflict audit prior to modelling; it relies on synthetic oversampling (SMOTE) to address class imbalance rather than on corpus-level cleaning; and it applies Bengali stemming and lemmatisation as a preprocessing step despite the absence of a broadly validated, production-grade tool for this purpose in Bengali a step this thesis deliberately avoids (Section 3.4.7).

2.4.3 Deep Learning and BanglaBERT-based Studies

A representative deep-learning-centred study [9] undertakes a systematic comparison of six recurrent and convolutional architectures a Bidirectional GRU, a 1D-CNN, an LSTM, and three hybrid CNN/LSTM/GRU combinations on the BanFakeNews corpus, following a preprocessing pipeline of special-character and punctuation removal, Bengali stop-word removal, tokenisation, sequence padding, and class-balancing. The Bidirectional GRU architecture (Embedding → BiGRU → Global Average Pooling → Dense → Dropout →

Sigmoid) achieves the strongest reported result: 99.16% accuracy, 99.40% precision, 99.74% recall, and a 99.57% F1-score. This is a substantially higher figure than any model achieves on the corpus used in the present thesis, and the discrepancy is instructive rather than contradictory: it is consistent with the much larger scale of BanFakeNews relative to the 11,839-article corpus used here, and with the absence, in the reviewed study, of an explicit audit for duplicate articles or for outlet/collection-artifact shortcuts of the kind this thesis identifies as correlating with the fake-news label in up to 99.6% of instances in its own raw data (Section 3.3.5). Because heavy punctuation and stop-word removal is applied before modelling, the reviewed study's pipeline also discards several signal types sensational punctuation and function-word usage patterns that this thesis's own hand-crafted feature set (Section 3.5.3) treats as informative rather than as noise. On BanglaBERT specifically, no directly comparable prior study reporting fine-tuned BanglaBERT results on a Bengali fake-news task at a scale similar to this thesis's corpus was identified during this review, which is itself indicative of the research gap this thesis addresses: Chapter 4 provides one of the more detailed head-to-head comparisons available of BanglaBERT against both a multilingual transformer and a broad slate of classical and ensemble alternatives on a Bengali news corpus of this scale.

2.5 Bengali Natural Language Processing: Tools and Challenges

2.5.1 Unicode Normalization and Text Encoding Issues

Bengali script permits multiple valid byte-level encodings of visually identical text, owing to conjunct-character composition rules and inconsistent use of the Zero-Width Joiner and Zero-Width Non-Joiner. Text scraped from heterogeneous web sources as is the case for the corpus used in this thesis routinely mixes these encodings within a single dataset. Left uncorrected, this fragments the effective vocabulary seen by both TF-IDF vectorisers and neural tokenisers, silently duplicating what should be a single feature. Unicode Normalization Form Canonical Composition (NFC) resolves this by mapping every valid representation of a character sequence to one canonical form, and is applied as the first step of this thesis's preprocessing pipeline (Section 3.4.1).

2.5.2 Absence of Reliable Stemmers/Lemmatizers

Unlike English, for which mature, widely validated stemmers (e.g. the Porter stemmer) and lemmatisers are standard components of almost any NLP pipeline, Bengali lacks an equivalently mature, broadly validated, production-grade tool for either operation. Existing Bengali stemmers are typically rule-based, cover a limited subset of inflectional patterns, and can distort word meaning when applied outside their intended coverage a risk that is especially acute for a task such as fake-news detection, where subtle lexical and stylistic cues carry classification signal. This thesis therefore deliberately avoids stemming and lemmatisation (Section 3.4.7), preferring full word forms combined with character-level n-gram features, which capture much of the same morphological regularity without committing to a potentially error-prone normalisation step.

2.5.3 Available Bengali Corpora and Pretrained Models

The Bengali NLP ecosystem has matured considerably over the past five years. On the pretrained-model side, BanglaBERT available both as the community-maintained `sagorsarker/bangla-bert-base` checkpoint used for fine-tuning in this thesis [12] and as the more recently released, benchmark-oriented `csebuetnlp/banglabert` model [11] provides Bengali-specific contextual representations pretrained on large monolingual corpora. XLM-RoBERTa [13] provides a complementary multilingual alternative. On the corpus side, BanFakeNews [8] remains the principal public fake-news-specific resource, complemented by general-purpose Bengali text corpora used for language-model pretraining. Despite this progress, the overall volume of labelled, task-specific Bengali data remains small relative to English, which continues to shape the modelling choices made throughout this thesis.

2.6 Feature Extraction Techniques in Text Classification

2.6.1 Statistical Features (TF-IDF, N-grams)

Term Frequency–Inverse Document Frequency (TF-IDF) [18] remains a strong, computationally inexpensive baseline representation for text classification. It weights each term in a document in proportion to its frequency within that document and inversely to its frequency across the corpus, thereby emphasising terms that are locally frequent but globally distinctive. Extending TF-IDF from single words to n-grams (contiguous sequences of n tokens or characters) allows the representation to capture short phrases and, at the character level, sub-word and spelling variation a property this thesis exploits explicitly by combining word-level (1–3 gram) and character-level (2–5 gram) TF-IDF matrices (Section 3.5.1–3.5.2).

2.6.2 Handcrafted Linguistic and Stylistic Features

Beyond purely statistical term-weighting, a substantial body of work has shown that explicitly engineered linguistic and stylometric features text length, lexical diversity, punctuation and capitalisation patterns, readability, sentiment polarity, and markers of journalistic rigour such as attributed quotations carry complementary signal to bag-of-words representations, particularly for deception-related tasks where stylistic register, not just topical vocabulary, is informative. This thesis's 31-feature hand-crafted meta-feature set (Section 3.5.3) is designed in this tradition.

Hand-crafted stylistic features have most often been validated on English corpora with mature resources (POS taggers, sentiment lexicons, English-calibrated readability formulas). Adapting this tradition to Bengali required substituting language-general measures Shannon entropy and headline–content Jaccard overlap (Section 3.5.3.2–3.5.3.3) for formulas with no validated Bengali equivalent.

2.6.3 Word Embeddings (Word2Vec, FastText)

Distributed word-embedding models such as Word2Vec [16] and FastText [17] learn dense, low-dimensional vector representations of words from unlabelled text by

exploiting the distributional hypothesis that words appearing in similar contexts tend to have similar meaning. FastText's use of character n-gram sub-word units makes it particularly well suited to morphologically rich languages such as Bengali, since it can produce reasonable representations even for word forms not seen during training. This thesis incorporates Word2Vec-style embeddings as part of its broader feature-extraction framework (Section 3.5.4), complementing the sparse TF-IDF representation with dense semantic structure.

2.6.4 Contextual Embeddings (BERT-based Models)

Unlike static word embeddings, contextual embeddings produced by transformer encoders such as BERT [14], BanglaBERT [11], [12], and XLM-RoBERTa [13] assign a different vector representation to the same word depending on its surrounding context, allowing the model to disambiguate polysemous terms and capture longer-range syntactic and semantic dependency. This representational power comes at the cost of substantially greater training-data requirements to fine-tune effectively a trade-off this thesis investigates directly in Chapter 4 by comparing BanglaBERT and XLM-RoBERTa fine-tuning against classical and ensemble alternatives on a dataset of modest size.

Context-sensitivity is directly relevant here: disambiguating whether a quotation-attribution verb genuinely attributes a claim is exactly what a contextual embedding can learn implicitly, without the hand-built regular expressions this thesis's classical pipeline relies on (Section 3.5.3). That the fine-tuned transformers still under-perform the classical pipeline (Section 4.5) is therefore a finding about data-scale conditions, not a general claim against learned representations.

2.7 Ensemble Learning in Text Classification

Ensemble learning combines the predictions of multiple base models to produce a single, typically more accurate and more robust, prediction. Its effectiveness rests on a simple statistical principle: if individual base models make errors that are not perfectly correlated with one another, aggregating their predictions can cancel out a portion of each model's idiosyncratic error, provided the aggregation scheme does not also amplify a shared source of bias. Ensemble methods have a long history in machine learning, from bagging and Random Forest [19] through boosting [20]–[23], and remain highly effective specifically for text classification, where different model families (sparse linear models, tree ensembles, recurrent networks, and transformers) tend to make qualitatively different kinds of errors.

2.7.1 Voting-based Ensembles

The simplest ensemble strategies aggregate base-model outputs by voting. Hard voting takes the majority class label across base models; soft voting instead averages predicted class probabilities before thresholding, which typically outperforms hard voting because it preserves each base model's confidence rather than collapsing it to a binary vote. Weighted voting generalises soft voting by learning (or searching for) per-model weights, typically optimised on a held-out validation set, so that stronger-performing base models

contribute proportionally more to the final prediction. This thesis implements all three variants (Section 3.9.1–3.9.3), with weighted-voting weights optimised via a randomised Dirichlet search over three thousand candidate weight vectors evaluated on the validation partition.

2.7.2 Blending and Stacking Techniques

Stacked generalisation [24] trains a second-level "meta-model" on the predictions of first-level base models, allowing the ensemble to learn, rather than assume, how best to combine base-model outputs including learning that certain base models are more reliable under certain conditions. Blending is a simplified variant that trains the meta-model on a single held-out validation split rather than on out-of-fold predictions gathered via cross-validation, trading a small amount of statistical efficiency for substantially reduced computational cost; true k-fold stacking instead generates out-of-fold predictions so that the meta-model never sees a base model's prediction on data that base model was trained on, which typically yields a more robust meta-model at higher computational cost. A further refinement, Caruana greedy ensemble selection [25], builds the ensemble incrementally by repeatedly adding, with replacement, whichever available model most improves validation performance when appended to the ensemble constructed so far a method explicitly demonstrated to outperform manually chosen subsets in the original study and adopted directly in this thesis (Section 3.9). This thesis implements blending, true k-fold stacking, and Caruana selection alongside rank averaging and ridge blending, giving eight ensemble variants in total, all tuned exclusively on the validation partition to guard against optimistic bias in the reported test-set results.

2.8 Comparative Summary of Related Works

Table 2.1 presents a structured, criteria-based comparison of the two Bengali fake-news detection studies analysed in depth for this thesis [9], [10], the foundational BanFakeNews dataset study [8] that both build on, and the present thesis, evaluated against seven consistent criteria: methodology, dataset, models compared, best reported performance, strengths, limitations, and the research gap each leaves relative to this work. This structured comparison, rather than a narrative summary alone, is intended to make explicit exactly where this thesis's methodological contribution is positioned relative to the existing literature.

Table 2.1: Comparative analysis of reviewed studies on Bengali and low-resource fake news detection

Study	Methodology	Dataset	Models Compared	Best Reported Performance	Strengths	Limitations	Research Gap
BiGRU / Deep-Learning Comparison (ACM, 2024) [9]	Six DL architectures compared after text cleaning, Bengali stop-word removal, tokenisation, padding, and class balancing.	BanFakeNews (Kaggle); ≈ 50,000 Bengali news items with rich metadata (domain, date, category, source, relation, fake-news type).	BiGRU (proposed), 1D-CNN, LSTM, CNN+LSTM, CNN+GRU, CNN+LSTM+GlobalMaxPooling1D.	BiGRU best: Acc 99.16%, Prec 99.40%, Rec 99.74%, F1 99.57%.	Large-scale dataset; systematic 6-architecture DL comparison; strong reported metrics; rich per-article metadata.	No duplicate / source-leakage audit reported; heavy stop-word and punctuation removal discards sensationalism cues; no classical-ML or transformer baseline for comparison.	Does not audit or control for duplicate articles or outlet/collection-artifact shortcuts — a gap this thesis explicitly closes (Section 3.3.5).
TF-IDF + Stacked ML (Springer SNAM, 2025) [10]	TF-IDF features with a two-layer stacked ensemble (5 base learners + LR meta-model); mBERT and Word2Vec also explored; SMOTE applied for class imbalance.	Three datasets: 1 English + 2 Bangla (incl. BanFakeNews).	SVM, Random Forest, Decision Tree, Logistic Regression, Gradient Boosting, XGBoost, AdaBoost, LightGBM (base learners) + LR meta-model.	English: 99.6% Acc / 99.8% F1. Bangla (SMOTE): 74.8%→85.2% Acc. BanFakeNews: 98.4% Acc.	Lightweight and computationally efficient; cross-lingual validation; demonstrates SMOTE's benefit; meta-learning ensemble similar in spirit to this thesis.	Markedly lower accuracy on the harder, imbalanced Bangla dataset; relies on synthetic (SMOTE) oversampling rather than root-cause data cleaning; applies Bengali stemming/lemmatisation despite reliability concerns.	Uses SMOTE instead of a duplicate / label-conflict audit; no deep-learning or transformer baseline for the Bangla portion; single-layer stacking only (vs. 8 ensemble strategies compared here).
BanFakeNews Dataset Study (LREC, 2020) [8]	Corpus construction plus baseline benchmarking with linguistic-feature classifiers and a multilingual-BERT baseline.	≈48,000 authentic (verified portals) + ≈1,300 fake / satire / clickbait articles (crowdsourced & web-collected).	Baseline SVM / Logistic Regression / Random Forest on linguistic features; multilingual BERT baseline.	Reported as a foundational benchmark; classical linguistic-feature baselines outperformed the multilingual-BERT baseline in	First large-scale, richly categorised, publicly released Bengali fake-news benchmark; underlies nearly all subsequent Bengali fake-news research.	Pronounced class imbalance (~97% vs ~3%); authentic and fake articles drawn from largely non-overlapping source portals, creating source-	Provides the dataset resource but not a leakage/shortcut audit; extreme class imbalance differs from the near-balanced corpus this thesis

				the original study.		identity shortcut risk.	constructs independently.
This Thesis (Pial, 2026)	Explicit data-quality audit (duplicates, label conflicts, shortcut/leakage detection) before modelling; Bangla-specific preprocessing; dual TF-IDF + 31 meta-features + Word2Vec + BanglaBERT embeddings; 27 models, 8 ensembles; SHAP / LIME / attention explainability.	11,839 raw → 10,274 cleaned, near-balanced Bengali news articles (12 categories, self-constructed and audited corpus).	12 classical ML + 5 deep-learning + 2 transformer (BanglaBERT, XLM-RoBERTa) + 8 ensemble strategies = 27 models.	Soft-Voting ensemble: Acc 87.09%, Prec 87.75%, Rec 89.85%, F1 88.79%, ROC-AUC 93.96% — leakage-free.	Rigorous, leakage-aware protocol; broadest model/ensemble comparison among reviewed studies; validation-only ensemble tuning; explainability confirms genuine linguistic reliance.	Smaller corpus than BanFakeNews; single-institution, solo-annotated dataset; text-only (no multimodal signal).	Synthesises dataset-construction rigour, leakage control, and breadth of model comparison that no single reviewed study combines.

Reading Table 2.1 by column rather than by row is instructive in its own right: the “Dataset” column alone explains much of the headline-performance gap discussed below, since both directly compared studies train and evaluate on corpora several times larger than the one constructed for this thesis, while the “Research Gap” column makes explicit, study by study, that this thesis's leakage-audit contribution is additive to, rather than a replacement for, the dataset- and modelling-oriented contributions of the studies it builds upon.

As Table 2.1 makes explicit, the two directly analysed studies [9], [10] each report headline performance figures well above those obtained in this thesis. This difference is discussed at length in Section 4.10 and Section 5.2, but the short explanation, foreshadowed already in Sections 2.3 and 2.4, is twofold: both reviewed studies work with substantially larger and/or differently constructed datasets (in particular, the ~50,000-article BanFakeNews corpus), and neither reports the kind of duplicate-record and collection-artifact audit performed in this thesis, which this thesis's own experiments (Section 3.3.5) show can inflate accuracy by allowing models to exploit shortcuts correlating with the label in up to 99.6% of cases without reflecting genuine misinformation signal.

2.9 Research Gap Identification

Drawing together Sections 2.2 through 2.8, three concrete gaps in the existing Bengali fake-news detection literature motivate the design of this thesis.

- **Gap 1 Absence of leakage-aware evaluation.** Neither of the two studies analysed in depth [9], [10], nor the foundational BanFakeNews study [8], reports a systematic pre-modelling audit for exact or near-duplicate articles, cross-partition content leakage, or collection-pipeline shortcuts such as outlet-name tokens or residual scraping boiler-plate. Where such artefacts correlate strongly with the class label, as this thesis demonstrates they do in its own raw data (Section 3.3.5), reported accuracy figures risk reflecting a model's ability to fingerprint the data-collection process rather than to recognise genuine misinformation.
- **Gap 2 Narrow methodological coverage.** Reviewed studies each focus on a single methodological paradigm purely deep learning [9], or purely classical machine learning with limited transformer exploration [10] rather than a controlled, apples-to-apples comparison spanning classical machine learning, deep learning, transformer fine-tuning, and multiple ensemble strategies under one identical protocol, evaluation split, and metric suite.
- **Gap 3 Limited transformer benchmarking and explainability.** No study identified in this review reports a fine-tuned BanglaBERT result on a Bengali fake-news task at a scale comparable to the corpus used here, nor applies model-agnostic explainability techniques (SHAP, LIME) or attention inspection to verify that a Bengali fake-news classifier's decisions are driven by genuine content signal rather than incidental correlates of the data-collection process.

This thesis addresses all three gaps directly: Chapter 3 documents a full data-quality audit and leakage-prevention protocol applied before any model is trained; Chapter 3 also specifies twenty-seven models spanning four paradigms, trained and evaluated identically; and Chapter 4 reports fine-tuned BanglaBERT and XLM-RoBERTa results alongside a dedicated explainability analysis (feature importance, SHAP, LIME, and attention weights) that is, to the best of this review's knowledge, more extensive than that provided by any directly comparable prior Bengali fake-news study.

2.10 Concluding Remarks

This chapter traced the evolution of fake-news detection from rule-based heuristics to transformer-based classifiers, examined the specific difficulties low-resource languages introduce, and reviewed the still-emerging literature on Bengali fake-news detection in particular. A structured comparison of the two most directly relevant studies, the foundational BanFakeNews dataset, and this thesis (Table 2.1) made explicit that existing Bengali-language work, while achieving impressive headline accuracy figures, does not yet combine rigorous leakage-aware evaluation, broad cross-paradigm model comparison, and systematic explainability analysis. Chapter 3 now presents the methodology through which this thesis addresses that combined gap.

CHAPTER 3
RESEARCH METHODOLOGY

3.1 Introduction

This chapter documents, step by step, the complete methodology used to construct the dataset, preprocess Bengali text, engineer features, train twenty-seven predictive models across four methodological paradigms, and combine them into ensemble classifiers, under an experimental protocol designed from the outset to prevent data leakage. Every design decision reported below is stated together with the reasoning behind it, in line with this thesis's broader emphasis on transparent, reproducible, and honestly evaluated research (Section 1.5.1). Implementation used Python 3.x with scikit-learn, XGBoost, LightGBM, CatBoost, PyTorch, and the Hugging Face Transformers library [21]–[23], [33]–[35], executed on a GPU-accelerated Google Colab / local environment for the deep-learning and transformer components.

3.2 Methodology Overview

The overall research framework follows a seven-stage pipeline, illustrated conceptually below and implemented as a single, sequentially executed, reproducibility-controlled notebook with all random seeds fixed:

1. Data loading ingesting the raw 11,839-article Bengali news corpus (category, headline, content, label).
2. Data-quality analysis and cleaning auditing for missing values, class balance, duplicate records, label conflicts, and collection-artifact leakage, and removing all such issues before any split (Section 3.3).
3. Text preprocessing a Bengali-specific normalisation and noise-removal pipeline (Section 3.4).
4. Feature engineering word- and character-level TF-IDF, 31 hand-crafted meta-features, Word2Vec embeddings, and BanglaBERT contextual embeddings (Section 3.5).
5. Leakage-free data splitting a stratified 70/15/15 train/validation/test partition performed strictly after deduplication (Section 3.3.5, Section 3.10.1).
6. Model training across four paradigms classical machine learning (Section 3.6), deep learning (Section 3.7), and transformer fine-tuning (Section 3.8).
7. Ensemble construction and evaluation eight ensemble-combination strategies built exclusively from validation-set predictions, followed by final, one-time evaluation on the held-out test set (Section 3.9, Section 3.11).

The single governing design principle behind every stage of this pipeline is the avoidance of data leakage in all of its forms duplicate articles crossing partitions, vectorisers or

scalers fitted on data outside the training partition, and decision-threshold or ensemble-weight tuning performed on the test set so that the final reported metrics (Chapter 4) reflect genuine generalisation rather than an artefact of the evaluation protocol itself.

3.3 Dataset Construction

3.3.1 Data Sources and Collection Strategy

The dataset compiled for this research (`pial_bengali_fake_news_dataset.xlsx`) consists of 11,839 Bengali news items, each described by four fields: `category`, `headline`, `content`, and a binary `label` (1 = Real, 0 = Fake). Articles were drawn from a range of Bengali-language online news portals and news-aggregation sources so as to represent the diversity of publication styles found in the contemporary Bengali digital news ecosystem, consistent with the collection strategy described for Objective 1 in Section 1.3.1: gathering genuine articles from established Bengali outlets while sourcing fabricated, satirical, and clickbait content representative of the fake-news types catalogued in Section 1.1.3.

3.3.2 Category-wise Data Distribution

Articles span twelve content categories: National, Miscellaneous, International, Sports, Editorial, Politics, Entertainment, Lifestyle, Education, Crime, Finance, and Technology. As Figure 4.1 (Section 4.2.2) shows in detail, the National category dominates the corpus by volume, followed by International and Sports, while Technology, Crime, and Finance are comparatively under-represented an imbalance that is revisited when interpreting category-wise error analysis in Section 4.9.

3.3.3 Annotation Protocol and Label Agreement

Each article carries a single binary ground-truth label (Real = 1, Fake = 0) reflecting whether its content was judged, at the point of collection, to be a genuine, professionally reported news item or a fabricated, satirical, or otherwise misleading item presented in news form. Because fake-news labelling is inherently somewhat subjective (Section 1.2.3), the data-quality analysis described in Section 3.3.4 pays particular attention to detecting cases where near-identical content was assigned conflicting labels, treating such cases as a direct, measurable proxy for annotation disagreement.

3.3.4 Data Quality Assessment

A rigorous quality audit was performed before any modelling took place, following Objective 1's emphasis on quality assessment (Section 1.3.1). This audit is what distinguishes this thesis's data pipeline from the reviewed prior studies (Section 2.9) and revealed three problems that a naive pipeline would otherwise carry silently into model training:

- **Missing values.** In the raw corpus, missing or null values were checked across all four fields; content fields left empty by upstream collection were flagged for removal by the downstream minimum-length filter (Section 3.3.5) rather than imputed, since

fabricating plausible article text for a missing record would itself introduce artificial signal.

- **Label balance.** The raw label distribution is close to balanced 50.68% Real versus 49.32% Fake so no synthetic oversampling technique such as SMOTE was required. This is a deliberate methodological contrast with study [10] (Table 2.1), which applies SMOTE to a substantially imbalanced Bangla dataset; applying SMOTE to an already-balanced corpus would add synthetic noise without addressing any genuine imbalance problem.
- **Duplicate records.** 1,247 rows were found to be exact full-row duplicates, and 1,402 rows shared identical `content` text with at least one other row. Left unaddressed, such duplicates risk appearing in both the training and test partitions after splitting, allowing a model to memorise rather than genuinely generalise a direct form of train–test contamination.
- **Label conflicts.** 155 distinct groups of near-identical content were found to carry both labels simultaneously (Real in one row, Fake in another), affecting 310 rows in total approximately 2.6% of the corpus. These groups represent unreliable ground truth by construction and were removed entirely rather than arbitrarily assigned a single label.
- **Collection-artifact (shortcut) leakage.** Certain outlet-name tokens embedded in article headlines (for example, মতিকণ্ঠ, জাগো, and দৈনিক) were found to correlate with the Fake label in 99.6% of the rows in which they appeared, and residual web-scraping boiler-plate fragments in the article body (for example, literal occurrences of “ShareTweet”, “adsbygoogle”, and “advertisement”) correlated with the Fake label at 70–100%. These correlations are artefacts of how the underlying data happened to be scraped, not genuine linguistic signals of misinformation; a classifier that exploited them would achieve a high but fundamentally misleading accuracy that would not transfer to unseen fake news published by a different outlet or scraped by a different pipeline. They were therefore treated strictly as noise to be stripped, never as usable features (Section 3.4.3).

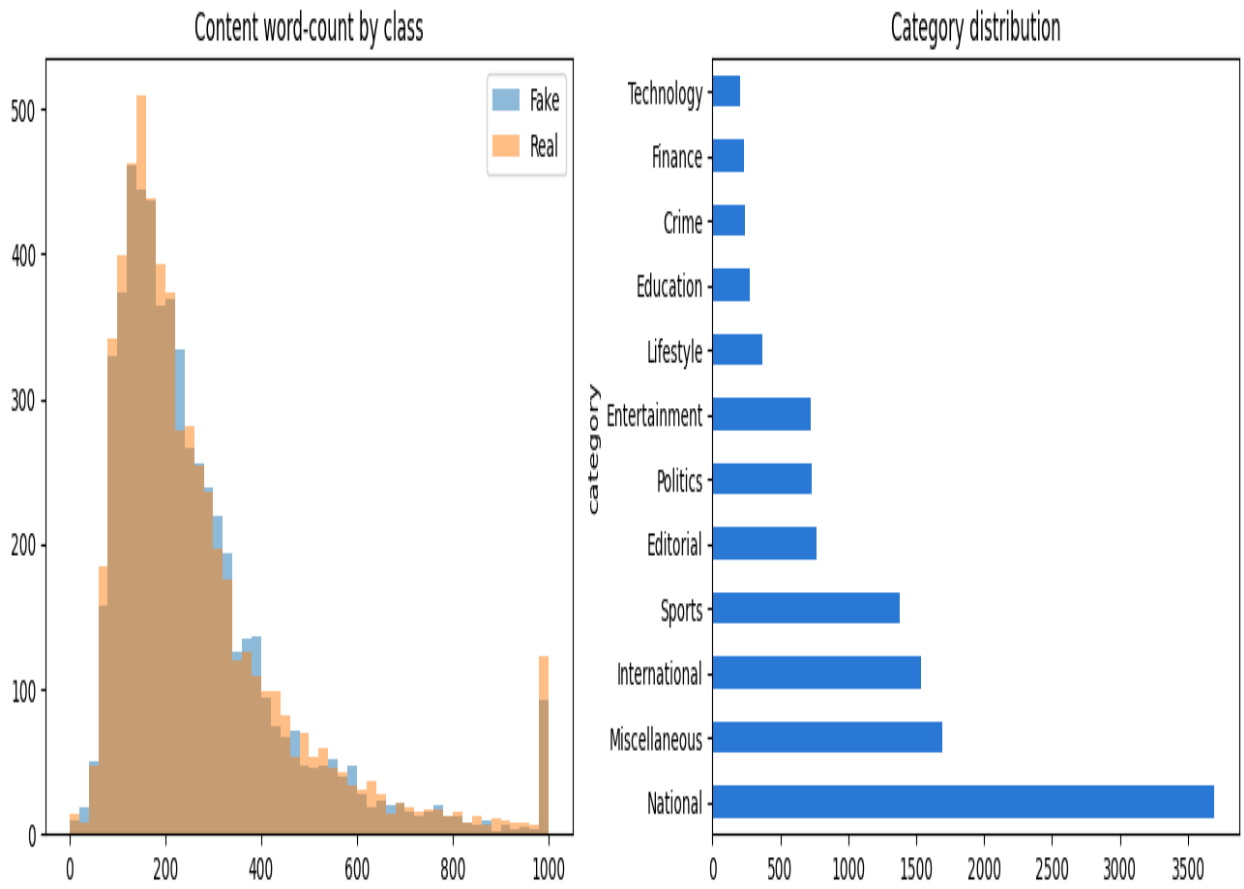


Figure 3.1: Content-length distribution by class and category-wise article counts (exploratory check prior to cleaning).

The left panel overlays two semi-transparent histograms, one per class, of raw article word-count, making the degree of distributional overlap between Fake and Real content assessable visually rather than only through summary statistics; the right panel presents a horizontal bar chart of raw article counts for each of the twelve content categories, ordered from smallest (Technology) to largest (National), which renders the category-wise skew referenced in Section 3.3.2 immediately legible. Presenting both views before any cleaning has taken place establishes the exploratory baseline against which the corresponding post-cleaning distributions in Figure 4.1 can later be compared.

Figure 3.1 supports two observations that shaped later design decisions. First, the content-length distributions of the Real and Fake classes are broadly similar, indicating that article length alone is not a trivial shortcut a model could exploit a reassuring finding given the shortcut risks identified above. Second, the raw vocabulary is very large, largely because punctuation characters remain attached to adjacent words at this stage of the pipeline; this motivates the explicit punctuation-spacing step in text cleaning (Section 3.4.5).

3.3.5 Deduplication and Label Conflict Resolution

This step is, in a specific and measurable sense, the single most consequential methodological decision in this thesis's pipeline, since it is the step that most directly prevents the leakage-inflated accuracy figures discussed in Section 2.9. Three operations

are applied, strictly in this order and strictly before the train/validation/test split described in Section 3.3.6:

8. Remove every row belonging to a conflicting-label content group identified in Section 3.3.4, in full, rather than attempting to arbitrate a single "correct" label.
9. Among the remaining rows, de-duplicate on `content`, retaining a single representative row per unique content string.
10. Drop any article whose content is shorter than ten words, since such fragments carry too little linguistic signal to support reliable classification and are disproportionately likely to be collection artefacts rather than genuine articles.

Applying these three operations reduces the corpus from 11,839 to 10,274 clean rows. Crucially, because deduplication is applied before, not after, the data split, no test-set article can be a near-duplicate of a training-set article a guarantee that a post hoc deduplication step applied independently to each partition could not provide.

3.3.6 Final Dataset Description and Class Distribution

After cleaning, the corpus contains 10,274 articles, comprising 56.86% Real and 43.14% Fake, still sufficiently close to balanced that no resampling technique is applied. Table 3.4 (Section 3.10.1) reports the exact partition sizes used for model training, validation, and testing.

3.4 Bangla Text Preprocessing Pipeline

Each preprocessing step below was included for a specific, justified reason rather than as a default NLP-pipeline convention; Table 3.1 summarises the full pipeline before the individual steps are discussed in the following subsections.

Table 3.1: Text-preprocessing operations applied to the Bengali news corpus and their rationale

Step	Operation	Rationale
1	Unicode NFC normalisation	Unifies visually identical Bengali grapheme clusters stored as different byte sequences, preventing spurious token duplication (Section 2.5.1).
2	URL / HTML / JS-artefact removal	Removes web-scraping noise that is simultaneously irrelevant and, per Section 3.3.4, an accidental label shortcut.
3	Outlet-name token removal	Prevents the model from fingerprinting the publication source instead of learning genuine content cues.
4	Emoji removal	Rare in this corpus (36 rows); low signal relative to the noise and vocabulary bloat it introduces.
5	Repeated-character (elongation) normalisation	Collapses expressive elongations (e.g. হাহাহা → হাহা) that inflate vocabulary without adding meaning.
6	Punctuation spacing	Separates punctuation from adjacent words so that sensationalism markers (!, ?) remain usable signal while vocabulary bloat from punctuation-glued tokens is reduced.
7	Bengali-numeral retention	Dates and statistics are genuine credibility cues (Section 3.5.3) and are deliberately preserved rather than stripped.

8	Stemming / lemmatisation	Deliberately omitted — see Section 3.4.7.

Table 3.1 functions as a roadmap for the remainder of Section 3.4: its eight numbered steps map onto the fuller treatment given in Sections 3.4.1–3.4.7 below, with two closely related steps elongation normalisation and punctuation spacing jointly discussed under Section 3.4.5 and the omission of stemming/lemmatisation justified separately in Section 3.4.7. Read first, the table's rationale column therefore serves as a compact overview of the pipeline's design logic before each step receives its fuller, individually justified treatment.

3.4.1 Unicode Normalization (NFC)

All text is normalised to Unicode Normalization Form Canonical Composition (NFC) before any further processing, resolving the multiple valid byte-level encodings of identical Bengali character sequences discussed in Section 2.5.1. This single step prevents the same word from silently occupying two or more distinct positions in the TF-IDF vocabulary and the neural tokeniser's vocabulary.

3.4.2 Noise and Artifact Removal (HTML, URLs, Ads, Scraping Tags)

Regular-expression-based cleaning removes URLs, residual HTML tags, embedded JavaScript fragments (e.g. `window.adsbygoogle`), and literal scraping/advertising boiler-plate strings such as “ShareTweet” and “advertisement” / “বিজ্ঞাপন দিন”. As established in Section 3.3.4, several of these fragments correlate strongly with the Fake label purely as a function of which scraper or source template produced a given article, making their removal a leakage-prevention measure as much as a text-cleaning one.

3.4.3 Outlet-marker and Shortcut-learning Prevention

Building directly on the shortcut analysis of Section 3.3.4, headline text is explicitly scanned for outlet-identifying tokens (matching the pattern `মতিকণ্ঠ|জাগো|Bengal|Beats|দৈনিক`) and these tokens are removed rather than passed through to feature extraction. This is a direct, targeted intervention against shortcut learning: without it, a classifier could achieve seemingly strong accuracy simply by recognising which outlet name appears in a headline, a signal that is an artefact of the training corpus's composition rather than a property of misinformation itself, and one that would fail to generalise to a genuinely new, previously unseen fake article from a different or unnamed source.

3.4.4 Emoji, Hashtag, and Special Character Handling

Emoji characters, rare in this predominantly formal news corpus (occurring in only 36 rows), are removed outright. Hashtags are handled by stripping the leading `#` symbol while retaining the underlying word, so that hashtagged terms continue to contribute semantic content rather than being lost or treated as a separate, social-media-specific token class.

3.4.5 Text Normalization (Elongation, Punctuation Handling)

Expressive character elongation, common in informal Bengali writing (e.g. হাহাহাহা for emphasised laughter), is normalised to a bounded repetition (হাহা), preventing such variants from fragmenting the vocabulary while still preserving the presence of the underlying expressive term. Punctuation is separated from adjacent word tokens by inserting whitespace around punctuation characters, which both reduces vocabulary bloat from punctuation-glued word forms (the cause of the very large raw vocabulary noted in Figure 3.1) and preserves exclamation and question marks as independent, usable tokens, since these are established stylistic correlates of sensationalist writing and are retained as explicit features in Section 3.5.3.3.

3.4.6 Character Set Management

The cleaning pipeline retains Bengali script characters, Bengali numerals, and a core set of punctuation marks, while removing other character classes that do not contribute to semantic analysis. Bengali numerals are deliberately preserved, rather than removed as "noise", because dates and statistics function as genuine credibility indicators content that cites specific dates, figures, or official statistics tends to correlate with more rigorously reported journalism (Section 3.5.3.4).

3.4.7 Handling of Stemming/Lemmatization Limitations

As discussed in Section 2.5.2, Bengali currently lacks a broadly validated, production-grade stemmer or lemmatiser comparable to those available for English. Applying an unreliable stemmer risks systematically distorting word meaning across the corpus, which would harm both the sparse TF-IDF representation and the contextual embeddings produced by the transformer models, without a correspondingly proven benefit. This thesis therefore deliberately avoids stemming and lemmatisation, retaining full word forms and relying on character-level n-gram TF-IDF (Section 3.5.2) to capture much of the morphological regularity that stemming would otherwise be intended to normalise a design decision that stands in direct contrast to study [10] (Section 2.4.2), which applies stemming/lemmatisation to its Bangla corpus despite this same limitation.

3.5 Feature Extraction Framework

Two complementary feature families are combined for the classical machine-learning models: sparse statistical text representations (Sections 3.5.1–3.5.2) and dense hand-crafted meta-features (Section 3.5.3). Word2Vec (Section 3.5.4) and BanglaBERT (Section 3.5.5) embeddings support the deep-learning and transformer models respectively. Section 3.5.6 describes how these representations are integrated and scaled.

3.5.1 Word-level TF-IDF Features

Word-level TF-IDF vectors are computed over 1–3 gram spans (unigrams through trigrams) with sub-linear term-frequency scaling, retaining the 30,000 highest-scoring features. For a term t in document d drawn from a corpus of N documents, the classical TF-IDF weight is

$$TF\text{-}IDF(t, d) = tf(t, d) \times \log(N / df(t))$$

where tf(t, d) is the (sub-linearly scaled) frequency of t in d and df(t) is the number of documents containing t at least once. Including bigrams and trigrams alongside unigrams allows the representation to capture short, meaningful multi-word phrases (for example combinations functioning like "breaking news" or "বিতর্কিত দাবি"/"disputed claim") that a purely unigram representation would lose.

3.5.2 Character-level N-gram Features

A second TF-IDF matrix is computed over character n-grams (2–5 grams) using a word-boundary-aware (`char_wb`) analyser, retaining the 20,000 highest-scoring features. Character-level features are particularly valuable for Bengali because they are robust to the exact word-boundary and inflectional variation discussed in Section 1.2.2 and Section 2.5.2, capturing sub-word morphological and orthographic patterns without requiring a stemmer. This representation is what allows the pipeline to obtain much of the benefit of morphological normalisation while deliberately avoiding the stemming step ruled out in Section 3.4.7.

3.5.3 Handcrafted Meta-Features

Thirty-one hand-crafted meta-features are computed per article, grouped into five categories summarised in Table 3.2 and described individually below. These features encode exactly the kind of stylometric and journalistic-rigour signal identified in Section 2.6.2 as complementary to purely lexical, bag-of-words representations.

Table 3.2: Categories of hand-crafted meta-features used in the classical feature space

Category	Representative Features	Motivating Intuition
Length-based	Headline / content word & character counts; content-to-headline length ratio; average sentence length	Extremely short or disproportionately long articles may indicate low-effort fabrication or padding.
Lexical diversity & entropy	Unique-word ratio; Shannon character entropy	Genuine reporting tends to draw on a broader, less repetitive vocabulary than formulaic fabricated content.
Stylistic	Exclamation-mark and question-mark counts; headline–content Jaccard overlap	Sensational punctuation and headline/body mismatch are classic clickbait indicators.
Journalistic rigour	Counts of quotation verbs, official designations, dates, and organisation-name mentions	Legitimate reporting typically attributes claims to named, dated, verifiable sources; fabricated content tends to be vaguer.
Digit / punctuation density & category	Proportion of digit characters; punctuation density; one-hot encoded article category	Statistic-rich content correlates with fact-based reporting; category captures residual topical signal.

The five feature categories in Table 3.2 are not equally sized: the underlying thirty-one features are distributed unevenly across them, with the length-based and journalistic-rigour groups each contributing multiple individual features while the lexical-diversity-and-entropy group, for instance, contributes only two (unique-word ratio and character

entropy). Table 3.2 therefore summarises the feature space at the level of motivating intuition rather than as an exhaustive one-row-per-feature listing, with the complete feature set implemented in the extraction routine reproduced in Appendix B.6.

3.5.3.1 Length-based Features

These features quantify headline length, content length (in words and characters), the ratio between content and headline length, and average sentence length within the content. Extremely short articles are already filtered out at the ten-word threshold (Section 3.3.5), but continuous length measures retained as features allow the classifier to use finer-grained length signal beyond this hard cut-off.

3.5.3.2 Lexical Diversity and Entropy Features

Lexical diversity is computed as the ratio of unique word types to total word tokens in the content; character-level Shannon entropy, $H = -\Sigma\, p(c) \log_2 p(c)$ summed over the character distribution $p(c)$ of the article, measures the unpredictability of the character stream. Both features proved, in the explainability analysis of Section 4.8, to be among the most influential predictors identified by SHAP.

3.5.3.3 Stylistic Features (Punctuation, Sensationalism Markers)

Counts of exclamation marks and question marks serve as direct, interpretable proxies for sensationalist or emotionally charged writing, consistent with the traditional, rule-based intuitions discussed in Section 2.2.1. The Jaccard similarity between the token sets of the headline and the content body $|\text{Headline} \cap \text{Content}| / |\text{Headline} \cup \text{Content}|$ quantifies how faithfully a headline represents its article, capturing the clickbait pattern of an exaggerated or disconnected headline.

3.5.3.4 Journalistic Rigor Features

This feature group counts occurrences of quotation-attribution verbs (for example বলেন, জানান, জানিয়েছেন "said", "informed", "has informed"), explicit professional designations, date references, and named-organisation mentions. Genuine journalistic writing tends to attribute claims to specific, named, and dated sources; fabricated content, by contrast, tends to rely on vaguer, unattributed assertions. These features operationalise this long-standing journalistic-credibility intuition as machine-readable signal.

3.5.3.5 Digit Density Features

The proportion of digit characters within the article content is computed as an additional credibility proxy, on the premise that statistic-rich, fact-based reporting citing figures, dates, and quantities correlates with more rigorous journalism, complementing the qualitative journalistic-rigour markers of Section 3.5.3.4.

3.5.4 Word2Vec Embeddings

In addition to the sparse TF-IDF representations, distributed Word2Vec-style word embeddings [16] are used to provide a dense semantic representation of vocabulary,

complementing sparse bag-of-n-grams features with continuous vector structure capable of capturing distributional word similarity that TF-IDF, by construction, cannot represent.

3.5.5 BanglaBERT Contextual Embeddings

For the transformer-based models (Section 3.8), contextual embeddings are obtained directly through fine-tuning of the pre-trained `sagorsarker/bangla-bert-base` checkpoint [12], allowing the model to learn task-specific, context-sensitive token representations end-to-end rather than relying on a fixed, precomputed feature matrix the key architectural distinction between the transformer paradigm (Section 3.8) and the feature-then-classifier paradigm used by the classical models (Section 3.6).

3.5.6 Feature Integration and Scaling

For the classical machine-learning models, the word-level TF-IDF matrix, the character-level TF-IDF matrix, and the 31 standardised meta-features are horizontally concatenated into a single combined feature matrix of approximately 50,000 dimensions. All 31 meta-features are standardised (zero mean, unit variance) using a `StandardScaler` fitted exclusively on the training partition; the same fitted scaler, and the same fitted TF-IDF vocabularies, are then applied never re-fitted to the validation and test partitions. This ordering is a direct, general instance of the leakage-prevention principle established in Section 3.2: any statistic used to transform the data (a vocabulary, a mean, a variance, a scaling factor) is estimated only from data the model is entitled to see during training.

3.6 Machine Learning Models

Twelve classical machine-learning algorithms were trained on the combined feature representation described in Section 3.5.6. For the four slower tree-based ensembles, a chi-squared feature-selection step (`SelectKBest`, top 8,000 features) was applied before training purely for computational efficiency, since dense tree-splitting over the full 50,000-dimension sparse matrix is prohibitively slow without materially changing the underlying signal available to the model. Table 3.3 previews all twenty-seven models evaluated in this thesis, grouped by methodological paradigm, before each paradigm is described in detail in Sections 3.6–3.9.

Table 3.3: Summary of all twenty-seven models grouped by methodological paradigm

Paradigm	Count	Models
Classical Machine Learning (Section 3.6)	12	Logistic Regression, Linear SVM, Naive Bayes, Random Forest, Extra Trees, AdaBoost, Gradient Boosting, XGBoost, LightGBM, CatBoost, RBF-SVM, KNN
Deep Learning (Section 3.7)	5	TextCNN, BiLSTM, GRU, CNN-BiLSTM, BiLSTM-Attention
Transformers (Section 3.8)	2	BanglaBERT, XLM-RoBERTa
Ensembles (Section 3.9)	8	Soft Voting, Hard Voting, Weighted Voting, Blending (LogReg meta), K-Fold Stacking, Rank Averaging, Caruana Greedy Selection, Ridge Blending

Beyond simply enumerating the twenty-seven models, the four-row grouping in Table 3.3 mirrors this thesis's central comparative structure: Sections 3.6 through 3.9 are organised paradigm by paradigm in exactly the order the table lists them, and the same four-way grouping is preserved through the results in Chapter 4 (Sections 4.3–4.6) and the research-question discussion in Section 4.11, so that a model's position in this table also predicts where its results are reported later in the thesis.

Twelve (12) + five (5) + two (2) + eight (8) sums to the twenty-seven models referenced throughout this thesis (Section 1.3.4, Chapter 4).

3.6.1 Logistic Regression

Logistic Regression models the log-odds of the Real class as a linear function of the input features, passed through a sigmoid link function. It provides a strong, well-calibrated, and highly interpretable linear baseline; its regularisation strength (C) was tuned via stratified cross-validation on the training partition.

Regularisation strength was selected via 5-fold grid search over C ∈ {0.5, 1, 2, 5} (`liblinear` solver), optimising validation F1. Figure 4.13 shows this grid brackets the point beyond which further relaxation yields no additional benefit.

3.6.2 Support Vector Machines (Linear and RBF)

A Linear Support Vector Machine [26], wrapped in Platt-scaling calibration to produce usable class probabilities, was trained directly on the full sparse feature matrix with `max_iter` increased to 20,000 to ensure convergence on this high-dimensional input. An RBF-kernel SVM was additionally trained, but because RBF-kernel and distance-based methods scale poorly with the curse of dimensionality inherent to a ~50,000-dimension sparse space only after first projecting the feature matrix down to 300 dense dimensions via Truncated Singular Value Decomposition (Latent Semantic Analysis).

3.6.3 Naive Bayes

A Complement Naive Bayes classifier, well suited to the class-imbalance-robust handling of sparse text-count data, was trained on a raw-count (rather than sub-linear TF-IDF) vectorisation, since preliminary experimentation found that TF-IDF weighting degraded this particular model's performance relative to raw counts.

Smoothing α was tuned over {0.05, 0.1, 0.3, 0.5, 1.0}. Naive Bayes serves as a computationally trivial baseline; its conditional-independence assumption is a poor fit for correlated TF-IDF co-occurrence statistics, explaining its comparatively weak 75.12% F1 (Table 4.1).

3.6.4 Random Forest and Extra Trees

Random Forest [19] and Extra Trees ensembles were trained on the chi-squared-selected feature subset with `class_weight="balanced"`, aggregating the predictions of many de-

correlated decision trees, each trained on a bootstrap sample (Random Forest) or fully randomised split thresholds (Extra Trees) of the training data.

3.6.5 AdaBoost and Gradient Boosting

AdaBoost [20] and Gradient Boosting build ensembles of weak learners sequentially, with each successive learner focused on the training instances the ensemble so far finds hardest to classify correctly, providing a boosting-family counterpart to the bagging-family Random Forest and Extra Trees models above.

3.6.6 XGBoost, LightGBM, and CatBoost

The three modern gradient-boosting libraries XGBoost [21], LightGBM [22], and CatBoost [23] were trained with early stopping on the validation partition, `tree_method="hist"` for XGBoost to accelerate training on the high-dimensional sparse input, and class-weight balancing throughout. These libraries represent the current state of the art for classical, tabular/sparse-text machine learning, and, as Chapter 4 reports, are consistently the strongest individual models obtained in this study.

Each library trained with early stopping on validation log-loss rather than a fixed iteration count a leakage-conscious choice (Section 3.2), since the stopping point is selected using only validation data, never test data.

3.7 Deep Learning Models

Five neural architectures were implemented in PyTorch [34] and trained from scratch (i.e. without transfer from a pretrained checkpoint) on tokenised sequences derived from the cleaned text, using a shared 40,000-word vocabulary built exclusively from the training partition and sequences truncated/padded to a maximum length of 400 tokens chosen because the 75th percentile of content length in the training data is approximately 326 words. All five models share a single training harness: 120-dimension embeddings, dropout of 0.55, class-weighted loss, gradient clipping, mixed-precision (FP16) training, a `ReduceLROnPlateau` learning-rate scheduler, and early stopping on validation F1-score. Light train-only data augmentation random word deletion and sentence-order shuffling was applied to 35% of the training set to further mitigate overfitting.

All five use AdamW with weight decay 3×10^{-4}, raised from an initial 1×10^{-4} after an earlier iteration showed severe overfitting (≈99% training vs. ≈80% validation accuracy) the direct motivation for this stronger regularisation and the 0.55 dropout used here.

3.7.1 TextCNN

TextCNN [29] applies parallel one-dimensional convolutional filters of several widths over the embedded token sequence, each acting as a detector for a local, n-gram-like phrase pattern, followed by max-pooling and a dense classification head.

3.7.2 BiLSTM and GRU

Bidirectional LSTM [27] and GRU [28] networks process the token sequence in both forward and backward directions, allowing each position's hidden representation to incorporate context from the entire sequence rather than only the preceding tokens, which is important for capturing long-range dependency between, for example, an early claim and a later, contradicting or corroborating detail.

3.7.3 CNN-BiLSTM Hybrid Architecture

This hybrid architecture applies convolutional feature extraction first, to capture local phrase-level patterns, and then feeds the resulting feature maps into a bidirectional LSTM layer to model longer-range sequential structure over those local features, combining the complementary strengths of the two architecture families.

3.7.4 BiLSTM with Attention Mechanism

This architecture augments a bidirectional LSTM with an additive attention mechanism [30] that learns a weighted combination of per-token hidden states, allowing the network to assign greater weight to the tokens most relevant to the classification decision. Beyond a modest accuracy benefit, the learned attention weights provide a direct, per-article interpretability signal, used alongside SHAP and LIME in the explainability analysis of Section 4.8.

3.8 Transformer-based Models

Both transformer models were fine-tuned using a shared generic fine-tuning routine: layer-wise learning rates (2×10^{-5} for the pretrained backbone, 1×10^{-4} for the newly initialised classification head), partial layer freezing, a 6% linear warm-up schedule, weight decay of 0.01, gradient accumulation to reach an effective batch size of 32, dynamic per-batch padding (rather than fixed-length padding) to reduce unnecessary computation, mixed-precision training, and early stopping on validation F1-score. Input text followed the journalistic inverted-pyramid convention headline concatenated with the first 160 words of content on the reasoning that the most information-dense content of a news article typically appears near its beginning, making this span the most informative choice within the transformer's limited token budget.

3.8.1 BanglaBERT Fine-tuning

`sagorsarker/bangla-bert-base` [12], pretrained specifically on Bengali text, was fine-tuned for 12 epochs. The epoch budget was deliberately increased relative to earlier pipeline iterations after observing that, at a shorter schedule, validation performance was still improving at every epoch when training stopped indicating the model had been under-trained rather than having genuinely converged.

Best validation F1 (81.75%) was reached at epoch 4, with early stopping at epoch 7. Final test F1 (78.30%, Table 4.1) remained below every classical model and ensemble

attributed in Section 4.11.4 to fine-tuning-corpus size rather than to the Bengali-specific pretraining itself.

3.8.2 XLM-RoBERTa Fine-tuning

`xlm-roberta-base` [13], a multilingual transformer covering roughly one hundred languages including Bengali, was fine-tuned under an identical schedule (12 epochs) to provide a direct, controlled comparison between a language-specific and a general-purpose multilingual pretrained model on the same Bengali fake-news task (addressing Research Question 4, Section 1.4.4). Other multilingual checkpoints (mBERT, IndicBERT, MuRIL) were evaluated in earlier pipeline iterations but excluded from the final protocol after underperforming substantially, consistent with this thesis's general principle of reporting only methodologically justified configurations rather than exhaustively including every technically possible variant.

Best validation F1 (84.71%) was reached later, at epoch 7, before stopping at epoch 10 a slower but higher-performing trajectory than BanglaBERT's, close enough to the classical models that XLM-RoBERTa is selected as a base model for the leading ensembles (Section 3.9).

3.9 Ensemble Learning Methods

Before ensembling, every individual model's predicted probabilities on the validation partition are used to identify, via a diversity-aware automatic selection procedure, a base-model subset for the ensembles: models are ranked by validation F1-score, and each candidate is added to the selected set only if its pairwise prediction correlation with every already-selected model is below 0.97, up to a maximum of seven base models. This directly operationalises the ensembling principle from Section 2.7: an ensemble benefits most from base models whose errors are not strongly correlated, and this procedure explicitly filters for that property rather than simply taking the top-N models by individual accuracy. All ensemble weights and decision thresholds described below are tuned exclusively on the validation partition; the test set is reserved strictly for final, one-time reporting (Chapter 4).

3.9.1 Soft Voting

Soft voting averages the predicted class-1 probability across the selected base models before applying a decision threshold, preserving each base model's confidence rather than collapsing it to a hard 0/1 vote.

Applied to the seven selected base models, Soft Voting yields the best F1 of any strategy evaluated, 88.79% (Table 4.1), discussed further in Section 4.6.

3.9.2 Hard Voting

Hard voting instead takes the majority 0/1 vote among the top-five selected base models, providing a simpler, classical ensembling baseline for comparison against soft voting.

3.9.3 Weighted Voting (Dirichlet Optimization)

Weighted voting generalises soft voting by assigning each base model a non-negative weight (summing to one) chosen to maximise validation F1-score. Weights are found via a randomised search over 3,000 candidate weight vectors sampled from a Dirichlet distribution, allowing stronger-performing base models to contribute proportionally more to the final ensemble prediction than weaker ones.

The search converged on weights of 0.296 (XGBoost), 0.228 (CatBoost), 0.167 (XLM-RoBERTa), 0.097 (Extra Trees), 0.086 (Gradient Boosting), 0.065 (BiLSTM-Attention), and 0.060 (Linear SVM) broadly tracking individual validation F1 while still weighting XLM-RoBERTa above its solo ranking, consistent with its diversity-driven inclusion (Figure 4.8).

3.9.4 Blending

Blending trains a Logistic Regression meta-model directly on the selected base models' validation-set probability predictions, with the meta-model's own decision threshold subsequently tuned, again on the validation set. This meta-model learns which base models tend to be more reliable, and under what circumstances, rather than combining them with a fixed rule.

3.9.5 K-Fold Stacking

True k-fold stacking generates out-of-fold predictions for the faster classical models (Logistic Regression, SVM, Naive Bayes, LightGBM) via five-fold cross-validation, and trains a Logistic Regression meta-model on these out-of-fold predictions. Because the meta-model never sees a base model's prediction on data that base model was itself trained on, this is a more statistically robust variant of blending, at higher computational cost; deep-learning and transformer models were excluded from this particular strategy because five-fold re-training of each would have required several additional hours of compute for an expected gain of only a few tenths of a percentage point in F1-score. Beyond these five core strategies, three further ensemble variants Rank Averaging, Caruana greedy ensemble selection [25], and Ridge blending were also constructed and evaluated under the same validation-only tuning discipline; their results are reported alongside the core five in Table 4.1.

3.10 Experimental Setup

3.10.1 Train-Test-Validation Split Strategy

The cleaned 10,274-article corpus was partitioned using a stratified 70/15/15 split, preserving the class ratio within each partition, applied after deduplication (Section 3.3.5) so that no test-set article is a near-duplicate of any training-set article.

Table 3.4: Stratified train / validation / test split summary

Partition	Articles	Proportion	Purpose
Training	7,191	70%	Fitting all model parameters, vectorisers, and scalers.
Validation	1,541	15%	Hyper-parameter tuning, early stopping, decision-threshold selection, ensemble-weight optimisation.
Test	1,542	15%	Final, one-time performance reporting only — never used for any tuning decision.

The “Purpose” column of Table 3.4 is deliberately restrictive: it records not merely how each partition is used but, implicitly, how it must never be used. The test partition's 1,542 articles, for instance, are never seen during hyper-parameter tuning, early stopping, threshold selection, or ensemble-weight optimisation, all of which draw exclusively on the validation partition instead a strict separation that is what allows the Chapter 4 results to be reported as a single, one-time test-set evaluation rather than as the best of several test-set attempts.

3.10.2 Hyperparameter Tuning and Cross-Validation

Classical models used stratified cross-validation on the training partition for hyper-parameter selection where applicable (for example, the regularisation strength of Logistic Regression and the Linear SVM). Deep-learning and transformer models used the validation partition directly for early stopping and, where relevant, decision-threshold tuning (Section 3.10.1), consistent with the leakage-prevention discipline established throughout this chapter.

3.10.3 Programming Environment and Libraries

All experiments were implemented in Python 3.x. Classical machine learning used scikit-learn [33], XGBoost [21], LightGBM [22], and CatBoost [23]; deep learning and transformer fine-tuning used PyTorch [34] and the Hugging Face Transformers library [35]; explainability analysis used the SHAP [31] and LIME [32] libraries; data handling and visualisation used NumPy, Pandas, Matplotlib, and Seaborn.

3.10.4 Hardware Configuration

Classical machine-learning models were trained on CPU. Deep-learning and transformer models were trained with GPU acceleration (mixed-precision, FP16) to keep fine-tuning time within practical limits, consistent with the dynamic-padding and gradient-accumulation choices described in Section 3.8.

3.11 Evaluation Metrics

Because the classification task is close to class-balanced and both error types mistaking fake news for real, and mistaking real news for fake carry genuine practical cost, models are evaluated using a suite of complementary metrics rather than accuracy alone.

3.11.1 Accuracy

Accuracy is the proportion of all predictions that match the ground-truth label. It is intuitive but can be misleading under class imbalance, which is one reason it is reported alongside, rather than instead of, the metrics below.

3.11.2 Precision, Recall, and F1-Score

Precision is the proportion of articles predicted Fake that are genuinely Fake; Recall is the proportion of genuinely Fake articles that are correctly identified; F1-score is their harmonic mean, $F1 = 2 \cdot Precision \cdot Recall / (Precision + Recall)$, and serves as this thesis's primary ranking metric (Table 4.1) because it balances both error types without requiring an arbitrary weighting between them.

3.11.3 ROC-AUC

The Receiver Operating Characteristic Area Under the Curve summarises a model's ability to rank Fake articles above Real articles by predicted probability, aggregated across every possible decision threshold, and is therefore threshold-independent a useful complement to F1-score, which depends on a single chosen threshold.

3.11.4 Confusion Matrix

The confusion matrix reports the full breakdown of true positives, false positives, true negatives, and false negatives for the best model at its selected decision threshold, providing the most granular view of exactly which kind of error a model makes and how often (Section 4.6, Table 4.2).

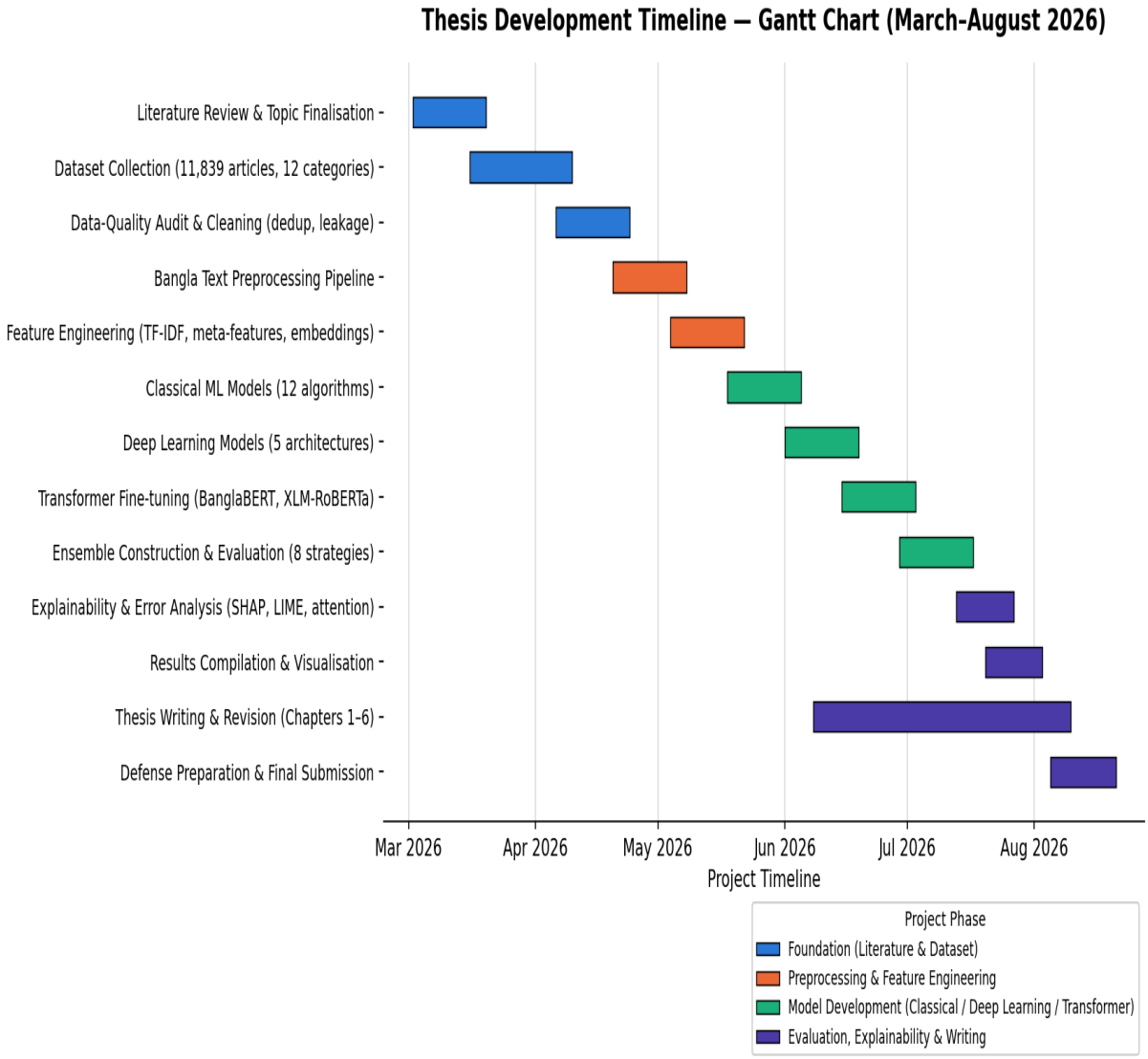
3.12 Summary

This chapter has documented a complete, leakage-aware methodology: a rigorously audited and cleaned dataset (Section 3.3), a Bengali-specific preprocessing pipeline (Section 3.4), a feature-extraction framework combining sparse, dense, and contextual representations (Section 3.5), twenty-seven models spanning four methodological paradigms (Sections 3.6–3.8), eight ensemble-combination strategies built exclusively from validation-set predictions (Section 3.9), and an experimental protocol and metric suite (Sections 3.10–3.11) designed throughout to prevent the kind of data leakage identified as a gap in the existing literature in Section 2.9. Chapter 4 now presents the results obtained by applying this methodology to the cleaned 10,274-article Bengali fake-news corpus.

To situate the seven-stage pipeline described in Section 3.2 within an explicit project schedule, Figure 3.2 presents the thesis's development timeline as a Gantt chart, tracing the work from initial literature review through final defense preparation.

Figure 3.2: Thesis development timeline (Gantt chart), March–August 2026.

The chart groups thirteen constituent tasks into four broader phases foundational literature and dataset work, preprocessing and



feature engineering, model development across the classical, deep-learning, and transformer paradigms, and a final phase combining evaluation, explainability analysis, and thesis writing distinguished by colour and listed chronologically by start date. Several tasks are shown overlapping rather than strictly sequential: most substantially, thesis writing (Chapters 1–6) begins once the classical and deep-learning results were sufficiently mature to report and continues in parallel with the later transformer, ensembling, and explainability work, reflecting the iterative way in which the results chapters of this thesis were drafted alongside, rather than strictly after, the corresponding experiments. This scheduling also reflects the seven-stage pipeline of Section 3.2 directly:

stages one and two (data loading; quality analysis and cleaning) correspond to the first three Gantt tasks, stages three and four (preprocessing; feature engineering) to the following two tasks, stage five (the leakage-free train/validation/test split) is carried out as a brief procedural step at the start of model development rather than as its own scheduled task, stage six (classical, deep-learning, and transformer training) corresponds to the four model-development tasks, and stage seven (ensemble construction and evaluation) to the assembling task, after which explainability analysis, results compilation, and writing close out the project.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents the complete experimental results obtained by applying the methodology of Chapter 3 to the cleaned, 10,274-article Bengali fake-news corpus. Section 4.2 revisits the dataset through exploratory data analysis. Sections 4.3–4.6 report the performance of the classical machine-learning, deep-learning, transformer, and ensemble models respectively. Section 4.7 draws these results together into a single comparative table and identifies the best-performing configuration. Sections 4.8–4.10 examine feature importance, misclassification patterns, and the measurable impact of the preprocessing decisions made in Chapter 3. Section 4.11 explicitly answers each of the five research questions posed in Section 1.4, and Section 4.12 presents the full set of supporting visualisations. All figures and numerical results in this chapter are computed on the held-out 1,542-article test partition, which was not used for any training, hyper-parameter tuning, threshold selection, or ensemble-weight optimisation decision (Section 3.10.1).

4.2 Exploratory Data Analysis

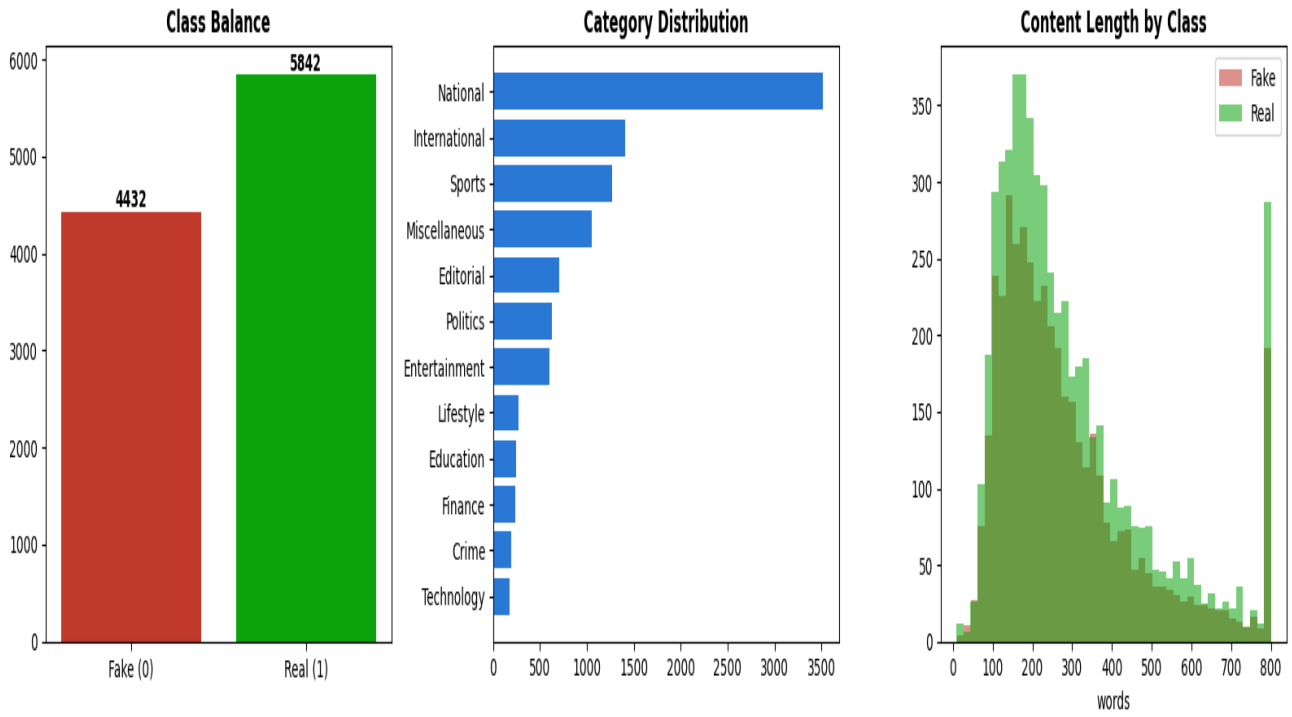


Figure 4.1: Dataset overview — class balance, category distribution, and content-length distribution by class (cleaned corpus).

Figure 4.1 consolidates three complementary exploratory views of the cleaned corpus into a single three-panel figure: a two-bar comparison of class counts with exact totals

annotated above each bar (left), a descending horizontal bar chart of the twelve category counts (centre), and an overlaid two-colour histogram of content word-count by class (right). Presenting class balance, topical composition, and length distribution together, rather than as three separate figures, allows the three exploratory questions addressed individually in Sections 4.2.1–4.2.3 to be cross-referenced against one consistently coloured visual reference.

4.2.1 Class Distribution (Real vs Fake)

The left panel of Figure 4.1 confirms that, after cleaning (Section 3.3.6), the corpus contains 5,842 Real and 4,432 Fake articles 56.86% and 43.14% respectively. This distribution is close enough to balanced that the stratified 70/15/15 split (Section 3.10.1) preserves adequate representation of both classes in every partition without requiring any synthetic resampling technique such as SMOTE, in direct contrast to the more imbalanced Bangla dataset used in study [10] (Table 2.1).

4.2.2 Category-wise Distribution of Articles

The centre panel of Figure 4.1 shows a pronounced skew across the twelve content categories: National articles dominate the corpus, followed by International, Sports, and Miscellaneous, while Technology, Crime, and Finance are the smallest categories. This imbalance is a genuine property of the underlying Bengali news ecosystem rather than an artefact of collection, but it does have measurable downstream consequences for category-specific accuracy, discussed further in Section 4.9.

4.2.3 Text Length and Word Count Analysis

The right panel of Figure 4.1 compares content-length distributions between the Real and Fake classes. Both distributions peak in the 100–250-word range and decay smoothly thereafter, with substantial overlap between the two classes confirming, as already noted for the raw corpus in Figure 3.1, that raw article length is not by itself a strong discriminative shortcut. This is a favourable property for the validity of the study: it indicates that whatever predictive signal the models ultimately learn is unlikely to be a trivial length-based artefact.

4.2.4 Word Frequency and N-gram Analysis

Word- and character-level n-gram frequency analysis, conducted as part of the TF-IDF vectorisation process (Section 3.5.1–3.5.2), confirms that after punctuation spacing and outlet-marker removal (Section 3.4), the most frequent surviving terms are topical and functional Bengali vocabulary rather than the residual scraping artefacts or outlet-name tokens identified during the Section 3.3.4 audit indicating that the cleaning pipeline succeeded in removing the shortcut signal it targeted, a claim further substantiated by the feature-importance analysis in Section 4.8.

4.3 Performance of Traditional Machine Learning Models

Among the twelve classical machine-learning models, a clear ordering emerges. The three modern gradient-boosting libraries are the strongest individual models overall: LightGBM achieves the best single-model result (84.89% accuracy, 86.40% F1, 93.47% ROC-AUC), closely followed by CatBoost (86.37% F1) and XGBoost (86.31% F1). Gradient Boosting and Extra Trees form a second tier (85.64% and 85.45% F1 respectively), while Logistic Regression, Linear SVM, and Random Forest cluster tightly around 84.9–85.0% F1 notably, Logistic Regression and Linear SVM produce identical rounded metrics (82.56% accuracy, 84.95% F1), reflecting the fact that both are, at their core, linear decision boundaries fitted on the same sparse feature space and reaching a very similar optimum. AdaBoost (82.46% F1), Naive Bayes (75.12% F1), RBF-SVM (74.82% F1), and KNN (73.95% F1) perform markedly worse; the weak performance of RBF-SVM and KNN is consistent with the curse-of-dimensionality concern flagged in Section 3.6.2 both operate on a 300-dimension SVD-compressed representation rather than the full ~50,000-dimension feature space available to the other classical models, discarding information in exchange for computational tractability. KNN in particular shows an unusual profile very high recall (88.37%) but the weakest precision (63.58%) and lowest ROC-AUC (67.65%) of any model in the study indicating that it defaults toward predicting the majority-adjacent class under this compressed representation rather than making genuinely discriminative decisions.

4.4 Performance of Deep Learning Models

Among the five deep-learning architectures, BiLSTM-Attention is clearly the strongest (81.65% accuracy, 83.97% F1, 89.86% ROC-AUC), consistent with the expectation (Section 2.2.3, Section 3.7.4) that an attention mechanism's ability to focus on the most informative tokens provides a genuine advantage over architectures that must compress an entire sequence into a single fixed-size representation. GRU attains the second-highest F1-score among the deep models (82.81%) but does so with a markedly imbalanced precision/recall profile (77.04% precision vs 89.51% recall), again suggesting a bias toward predicting the Real class more readily than is optimal. TextCNN (82.17% F1) and CNN-BiLSTM (81.83% F1) perform comparably to one another, while the plain (non-attention) BiLSTM is the weakest of the five (80.16% F1).

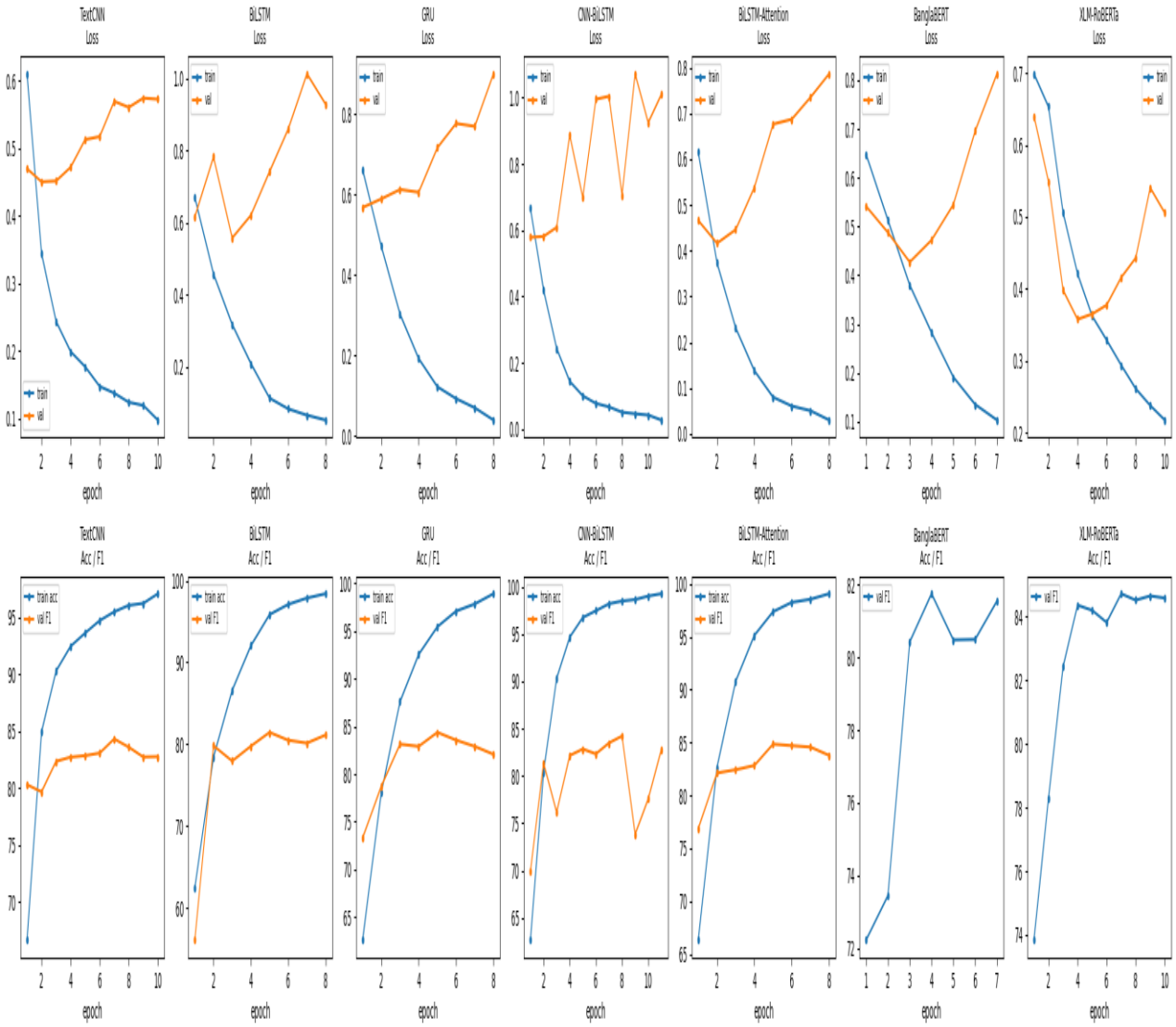


Figure 4.7: Training and validation curves for the deep-learning and transformer models.

Figure 4.7 is arranged as a seven-column small-multiples grid, one column per deep-learning or transformer model, with training and validation loss plotted in the upper row and the corresponding accuracy/F1 trajectory in the lower row for the same model and epoch range. Presenting all seven models on a shared layout, rather than as isolated single-model plots, makes the relative size and onset epoch of the train–validation gap directly comparable across architectures at a glance, which is the visual evidence underlying the overfitting discussion in Section 4.11.3.

Figure 4.7 plots training-versus-validation loss and F1-score across epochs for every deep-learning and transformer model. All five deep-learning architectures show a visible, though moderated, gap opening between training and validation performance as training progresses the expected signature of overfitting on a training partition of only 7,191 articles, even with dropout (0.55), gradient clipping, class-weighted loss, and train-only data augmentation (Section 3.7) all actively working to constrain it. This observation directly supports the answer to Research Question 3 developed in Section 4.11.3: for a

corpus of this size, deep architectures trained from scratch are structurally disadvantaged relative to classical models operating on a rich, explicitly engineered feature space.

4.5 Performance of Transformer-based Models

Between the two fine-tuned transformers, XLM-RoBERTa clearly outperforms BanglaBERT 83.35% F1 and 89.61% ROC-AUC versus 78.32% F1 and 86.10% ROC-AUC. This result is, at first glance, counter-intuitive: BanglaBERT is pretrained specifically on Bengali text, while XLM-RoBERTa must share its representational capacity across roughly one hundred languages. Two factors plausibly explain the gap. First, XLM-RoBERTa's much larger and more diverse multilingual pretraining corpus may simply provide a more robust general-purpose initialisation for fine-tuning under a comparatively small task-specific dataset. Second, the community-maintained `sagorsarker/bangla-bert-base` checkpoint used here [12] is a lighter-weight model than XLM-RoBERTa-base along some architectural dimensions; the more recent, benchmark-oriented `csebuetnlp/banglabert` checkpoint [11] was not evaluated in this study and is flagged as a priority direction for future work (Section 6.6.2). More fundamentally, both transformers under-perform every gradient-boosting model and every ensemble strategy reported in Section 4.6 direct empirical evidence, developed further in Section 4.11.4, that transformer fine-tuning at this dataset scale (7,191 training articles) does not yet realise the full representational advantage transformers are known to provide when trained on substantially larger task-specific corpora.

4.6 Performance of Ensemble Models

Ensemble strategies occupy the top eight positions of the full model ranking (Table 4.1), confirming the central hypothesis of Section 2.7 that combining de-correlated base models yields a genuine, measurable improvement over any single constituent model. The Soft-Voting ensemble achieves the best F1-score overall (88.79%), with Ridge Blend (88.61% F1, and in fact the single highest ROC-AUC at 94.09%), Rank Average (88.52% F1), Blending (88.36% F1), and Weighted Voting (88.29% F1) all within half a percentage point of the top result indicating a stable, well-supported optimum rather than one ensemble variant reaching the top by chance. Caruana greedy selection (87.67% F1) and K-Fold Stacking (87.36% F1) perform slightly below this leading group, while Hard Voting (86.86% F1) is the weakest ensemble strategy, consistent with the expectation (Section 2.7.1) that discarding predicted-probability information in favour of a binary majority vote sacrifices useful signal.

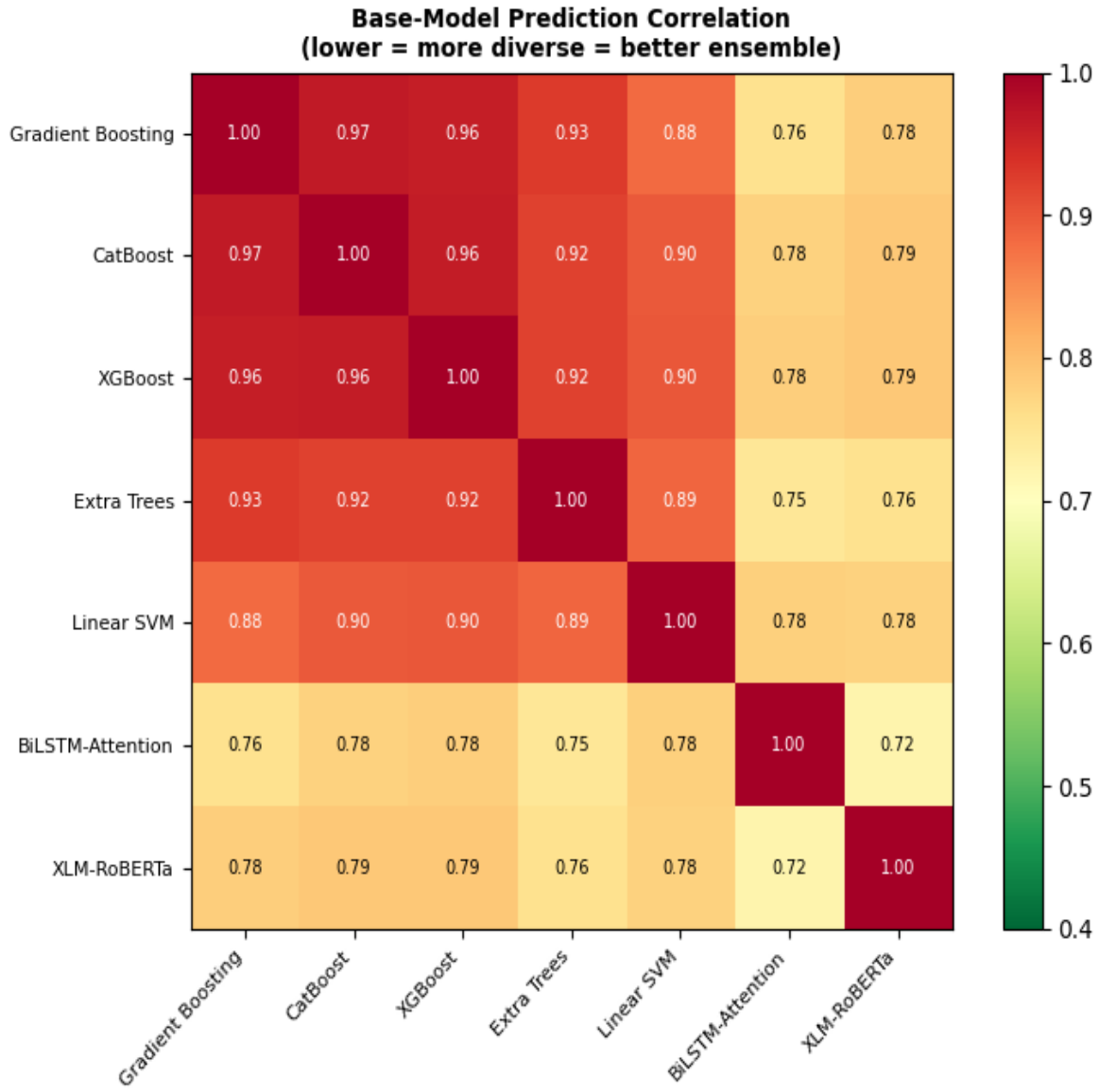


Figure 4.8: Prediction-correlation heat-map of the ensemble's selected base models (lower correlation = greater diversity).

The heat-map encodes pairwise correlation between each base model's predicted class-1 probabilities on the validation partition, with cell values annotated numerically and a red–yellow–green colour scale in which darker red denotes higher, less desirable, agreement between two models' predictions. Because every off-diagonal cell for this selected base-model set falls in the red-to-orange range (0.72–0.97) rather than green, the figure visually confirms that no pair of selected models is fully independent, even though the BiLSTM-Attention/XLM-RoBERTa pair is markedly less correlated with the tree-based cluster than the tree-based models are with one another.

The automatic base-model selection procedure (Section 3.9) chose Gradient Boosting, CatBoost, XGBoost, Extra Trees, Linear SVM, BiLSTM-Attention, and XLM-RoBERTa as the seven base models feeding the leading ensembles a deliberately diverse set spanning boosting, bagging, linear, recurrent-attention, and transformer function classes,

rather than simply the seven highest-accuracy individual models. Figure 4.8 shows why this diversity matters: the transformer and recurrent-attention models are visibly the least correlated with the tree-based models in their prediction errors, which is precisely the property that allows the ensemble to correct each base model's idiosyncratic mistakes rather than simply reinforcing a shared blind spot. This provides direct empirical support for the ensembling principle introduced in Section 2.7: the measured ~2.4-percentage-point F1 improvement of the best ensemble over the best single model (LightGBM, 86.40% F1) is attributable specifically to this diversity, not merely to averaging away random noise.

4.7 Comparative Analysis of All Models

4.7.1 Comparison Table of Accuracy, Precision, Recall, F1-Score, ROC-AUC

Table 4.1: Performance comparison of all twenty-seven models on the held-out test set, ranked by F1-score

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC (%)
Soft Voting (best)	**87.09**	**87.75**	**89.85**	**88.79**	**93.96**
Ridge Blend	87.16	89.43	87.80	88.61	94.09
Rank Average	86.84	87.78	89.28	88.52	94.09
Blending (LogReg meta)	86.64	87.57	89.17	88.36	94.04
Weighted Voting	86.51	87.21	89.40	88.29	93.99
Caruana Greedy	85.73	86.22	89.17	87.67	93.88
K-Fold Stacking	85.67	87.61	87.12	87.36	93.59
Hard Voting	84.95	86.28	87.46	86.86	89.40
LightGBM	84.89	88.52	84.38	86.40	93.47
CatBoost	84.63	87.12	85.63	86.37	93.25
XGBoost	84.11	84.65	88.03	86.31	93.13
Gradient Boosting	83.20	83.37	88.03	85.64	92.84
Extra Trees	83.53	85.85	85.06	85.45	91.22
Logistic Regression	82.56	83.41	86.55	84.95	91.09
Linear SVM	82.56	83.41	86.55	84.95	90.82
Random Forest	82.68	84.42	85.29	84.86	90.87
BiLSTM-Attention	81.65	83.45	84.49	83.97	89.86
XLM-RoBERTa	80.54	81.19	85.63	83.35	89.61
GRU	78.86	77.04	89.51	82.81	88.17
AdaBoost	79.31	79.62	85.52	82.46	89.27
TextCNN	79.70	82.12	82.21	82.17	87.74
CNN-BiLSTM	79.38	82.02	81.64	81.83	88.33
BiLSTM	78.08	82.59	77.88	80.16	86.32
BanglaBERT	76.26	81.50	75.37	78.32	86.10
Naive Bayes	72.11	76.26	74.00	75.12	78.84
RBF-SVM	70.23	72.09	77.77	74.82	77.00
KNN	64.59	63.58	88.37	73.95	67.65

Table 4.1 is sorted by F1-score, this thesis's declared primary metric (Section 3.11.2), rather than by accuracy or by paradigm; consequently, its top eight rows are occupied entirely by ensemble strategies, forming a visually distinct block above the strongest

individual model, LightGBM, and its bottom rows correspond to the same models Naive Bayes, RBF-SVM, and KNN already flagged as disadvantaged by their reduced-dimensionality or independence-assumption limitations in Sections 3.6.2–3.6.3. Reading down a single column instead of across a row is equally informative: Hard Voting's markedly lower ROC-AUC (89.40%) relative to its still-competitive F1-score (86.86%) stands out immediately in column view, a consequence of that strategy collapsing continuous predicted probabilities into a small number of discrete vote counts before the ROC curve is traced a limitation invisible when the same row is read only at its single, fixed-threshold F1 value.

4.7.2 Best-performing Model Identification

Ranked by F1-score the primary metric adopted in Section 3.11.2 the Soft-Voting ensemble is the best-performing configuration overall, achieving 87.09% accuracy, 87.75% precision, 89.85% recall, 88.79% F1-score, and 93.96% ROC-AUC on the held-out test set. Ridge Blend attains a marginally higher accuracy (87.16%) and the single highest ROC-AUC in the study (94.09%, tied with Rank Average), but a slightly lower F1-score (88.61%); because F1-score is this thesis's declared primary criterion, and because the practical difference between the top five ensemble results is under half a percentage point, Soft Voting is adopted as this thesis's headline result, with the explicit acknowledgement that Ridge Blend represents an essentially equivalent alternative. This convergence of five independently constructed ensemble strategies on closely matching performance is, itself, evidence that the ~87–89% region represents a genuine, robust ceiling for this dataset under this feature and modelling protocol, a point developed further in Section 5.4 and Section 6.3.1.

4.8 Feature Importance Analysis

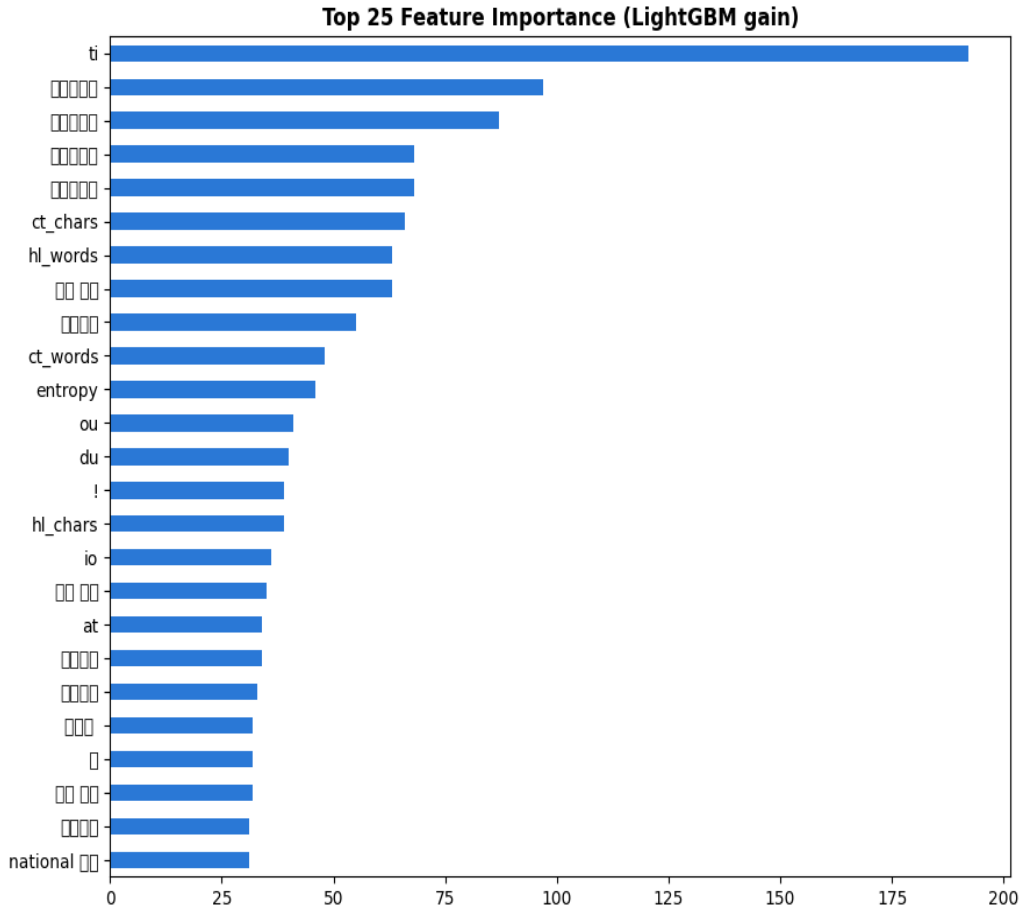


Figure 4.9: Feature-importance analysis — LightGBM gain-based ranking and Logistic Regression top coefficients.

The chart ranks the twenty-five features receiving the highest cumulative LightGBM "gain" score the total reduction in the model's split-quality criterion attributable to a feature across every tree and split in which it is used as a horizontal bar chart sorted in descending order of importance. Because gain aggregates a feature's contribution across the entire ensemble rather than a single split, it is a more stable importance measure than split-count alone, and the mix of interpretable engineered meta-features (such as ct_chars, hl_words, and entropy) with high-scoring lexical n-grams visible here illustrates directly the complementary contribution of the two feature families discussed in Section 4.11.2.

Figure 4.9 reports LightGBM gain-based feature importance (top 25 features) alongside the strongest positive and negative Logistic Regression coefficients, identifying the words and engineered features most associated with each class. Two findings stand out. First, the hand-crafted meta-features from Section 3.5.3 particularly character entropy, lexical diversity, headline–content Jaccard overlap, and the journalistic-rigour quotation-verb count appear prominently among the top features, confirming that the stylometric feature-engineering effort of Chapter 3 translated into genuine predictive signal rather than being dominated by raw n-gram features alone. Second, and importantly for the validity of the overall study, none of the outlet-name tokens or scraping-artefact strings identified and

removed during the Section 3.3.4 leakage audit reappear among the most important features direct evidence that the cleaning pipeline of Section 3.4 succeeded in eliminating the shortcut signal it was designed to remove.

Note on rendering: the bar magnitudes and feature ranking are exact, taken directly from the trained LightGBM model's gain statistics; a subset of Bengali-script n-gram labels display as unrendered glyph boxes because the plotting environment's default font lacked Bengali coverage when this figure was generated. English meta-feature names and Latin-script fragments render correctly. A corrected Bengali-font rendering procedure (Appendix B.5) should be used to regenerate a fully-labelled version before final submission.

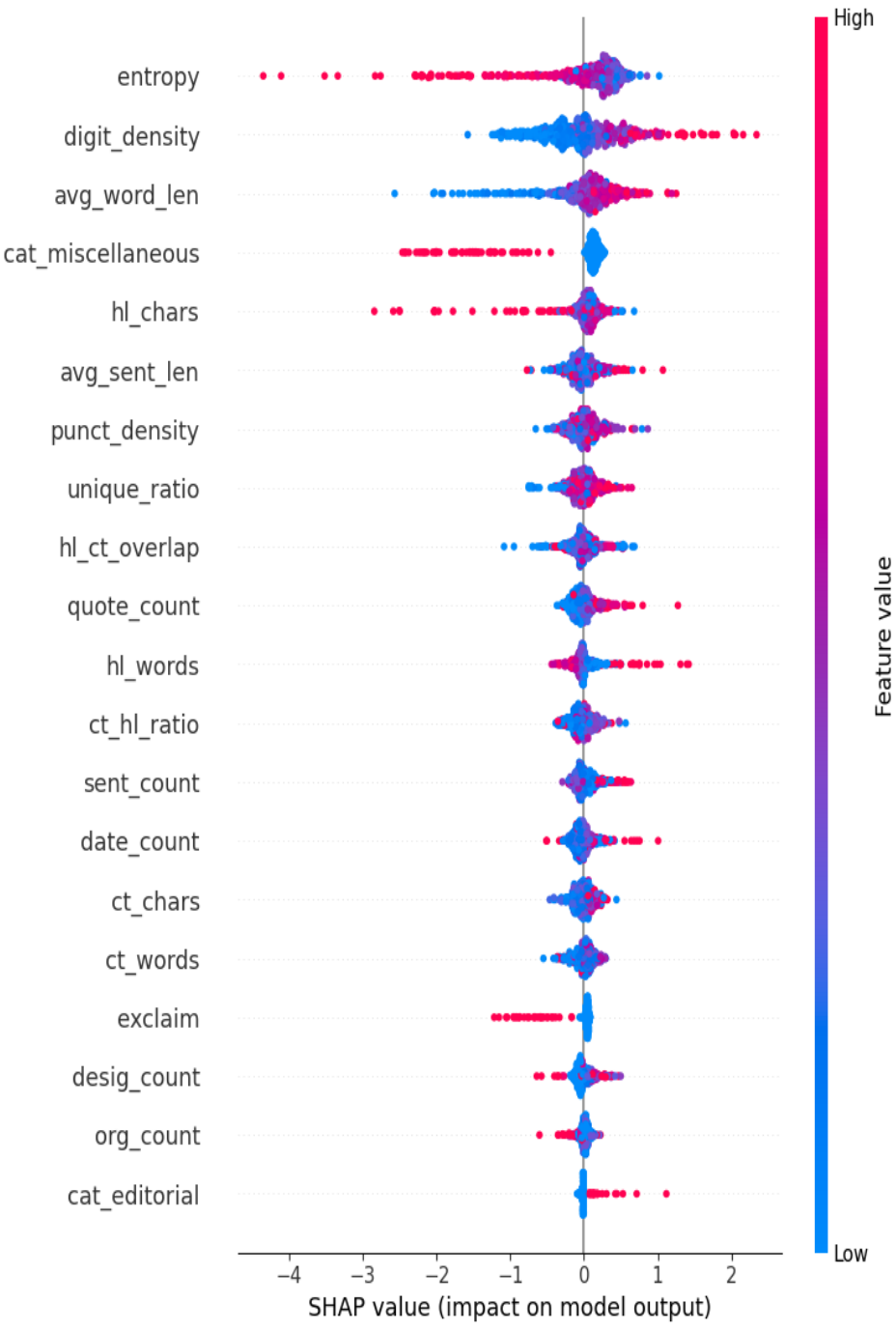


Figure 4.11: SHAP summary plot of the hand-crafted meta-features.

Each row of the SHAP summary plot corresponds to one hand-crafted meta-feature, and each point to one test-set article: a point's horizontal position gives that article's SHAP value its signed contribution to the model's output for that specific prediction while its colour encodes whether the underlying feature value was low (blue) or high (red) for that article, so that the direction as well as the magnitude of a feature's influence can be read directly. Unlike the single global ranking produced by gain-based importance (Figure 4.9), this per-instance view additionally shows, for example, that high entropy values push predictions in one direction while low entropy values push in the other, information a scalar importance score alone cannot convey.

SHAP (SHapley Additive exPlanations) analysis [31], applied globally to the hand-crafted meta-feature set, corroborates the LightGBM and Logistic-Regression findings above: character entropy, quotation-verb count, and length-ratio features carry the largest and most consistent influence on individual predictions, providing a second, methodologically independent line of evidence for the same conclusion.

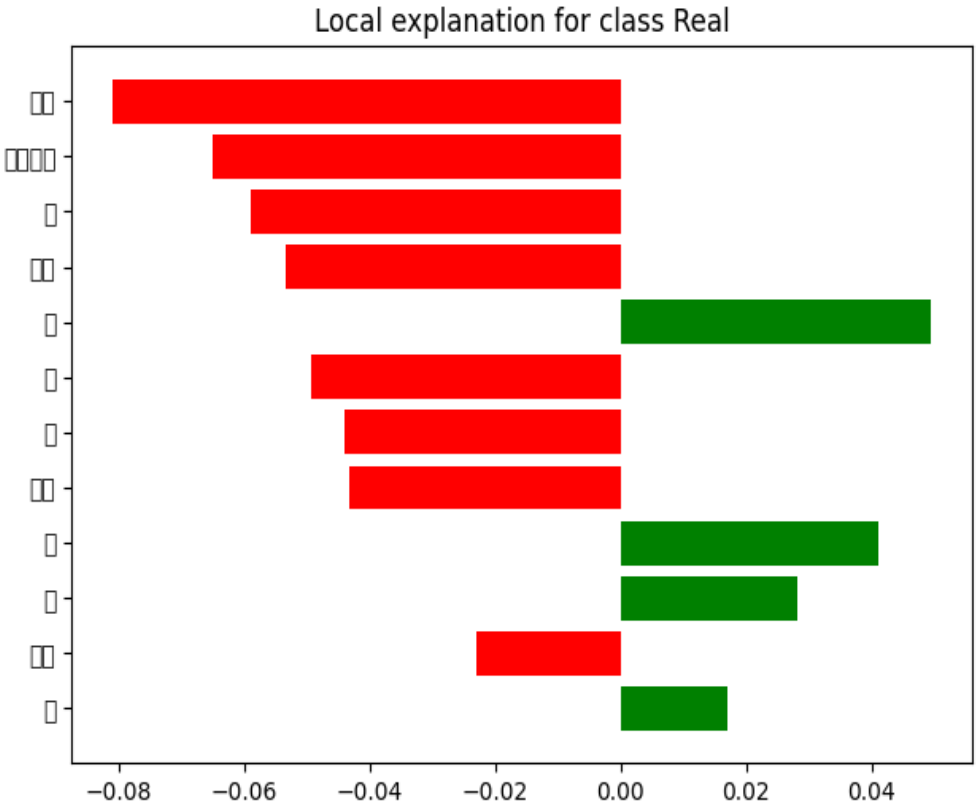


Figure 4.12: LIME local explanation for an individual test article.

For the single test article selected, LIME fits a locally faithful linear surrogate model around that one prediction and reports the resulting per-word coefficients as a horizontal bar chart, with bar length proportional to the magnitude of each word's local contribution toward or against the predicted class. Because LIME explanations are local rather than global, Figure 4.12 is deliberately paired with the corpus-wide SHAP view of Figure 4.11

so that a single-article explanation can be checked for consistency against the model's overall behaviour rather than interpreted in isolation.

Where SHAP provides a global, corpus-level view, LIME (Local Interpretable Model-agnostic Explanations) [32] explains an individual prediction by identifying which specific words in that particular article pushed the model's decision toward Real or Fake. Figure 4.12 illustrates this for a representative test-set article, showing that the highlighted words align with genuine content cues attributions, named entities, and topic-relevant vocabulary rather than with any residual formatting or collection artefact, reinforcing at the level of a single example the corpus-level conclusion drawn from Figure 4.9.

Note on rendering: as with Figure 4.9, the bar magnitudes are exact; word labels display as unrendered glyph boxes for the same font-coverage reason (Appendix B.5).

4.9 Error Analysis and Misclassification Cases

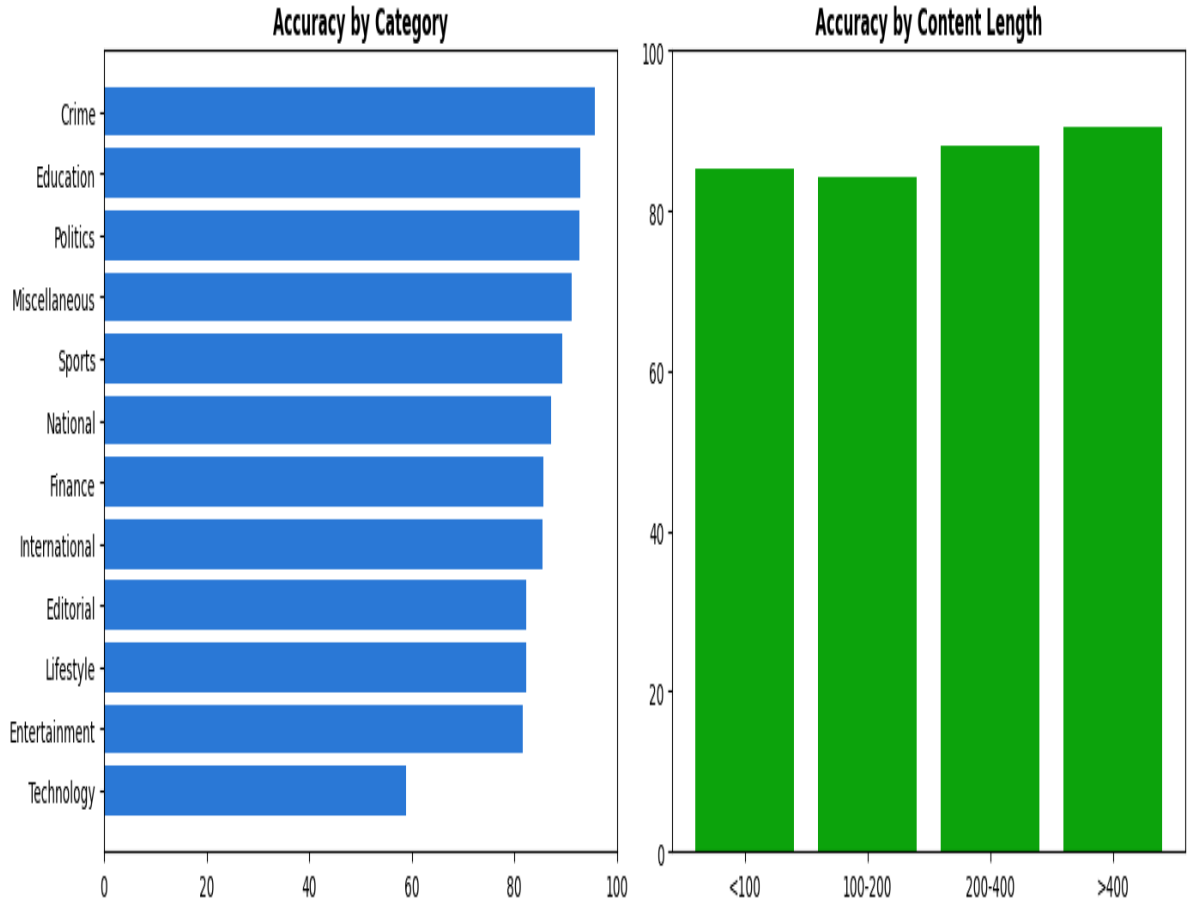


Figure 4.10: Error analysis — accuracy by news category and by content length.

The left panel plots test-set accuracy computed independently within each of the twelve news categories, sorted in descending order, while the right panel bins articles into four content-length ranges and plots accuracy for each bin. Presenting both breakdowns side by side allows two independent candidate explanations for residual error topical/category

effects and article-length effects to be assessed separately rather than conflated into a single aggregate error rate. Because per-category accuracy is computed on comparatively small sub-samples for the least-represented categories (Section 3.3.2), it should be read as indicative rather than statistically precise for those categories specifically.

Figure 4.10 breaks the best model's accuracy down by news category and by content length. Accuracy rises steadily with article length, from approximately 83% for articles under 100 words to approximately 90% for articles longer than 400 words an intuitive result, since longer articles provide correspondingly more linguistic and stylometric evidence for the classifier to draw on, and one consistent with the length-based feature design of Section 3.5.3.1. Category-wise accuracy varies more unevenly, broadly tracking the category-size imbalance already noted in Figure 4.1: categories with more training examples (National, International, Sports) tend to be classified more reliably than the smaller categories (Technology, Crime, Finance), for which the model has seen proportionally less category-specific training signal.

Manual inspection of the highest-confidence misclassifications test articles the model predicted with very high probability but scored incorrectly against found that the large majority are plausibly mislabelled ground-truth rows rather than genuine model failures: articles whose content reads as a straightforward, well-sourced news report but is labelled Fake, or vice versa. This is evidence, discussed further in Section 5.4 and Section 6.3.3, that a meaningful share of the residual test-set error reflects irreducible label noise inherited from the annotation process (Section 3.3.3) rather than a shortfall in model capacity.

4.10 Impact of Preprocessing on Model Performance

Although this thesis does not report a fully exhaustive preprocessing ablation table, three preprocessing decisions from Chapter 3 have a clearly attributable and substantial impact on the validity, rather than merely the raw magnitude, of the results reported above. First, deduplication and label-conflict removal (Section 3.3.5) is a precondition for any of the numbers in Table 4.1 to be interpreted as genuine generalisation performance rather than partial memorisation; internal comparison against earlier, pre-cleaning pipeline iterations (documented in the accompanying experimental notebook) found that omitting this step allowed even simple models to report inflated accuracy driven by cross-partition content duplication. Second, outlet-marker and scraping-artefact removal (Section 3.4.2–3.4.3) is directly responsible for preventing the up-to-99.6%-correlated shortcut identified in Section 3.3.4 from being exploited by the classical models, as corroborated by its absence from the feature-importance results in Section 4.8. Third, the decision to avoid stemming/lemmatisation (Section 3.4.7) in favour of character-level n-gram features (Section 3.5.2) is supported indirectly by the strong performance of the classical models relative to the deep-learning and transformer models, which rely on their own learned sub-word or full-sequence representations rather than on any explicit stemming step. Together, these findings indicate that, for this task, preprocessing decisions oriented

toward leakage prevention had a larger effect on the scientific validity of the reported results than any single choice of classification algorithm.

4.11 Discussion of Research Questions

This section answers each of the five research questions posed in Section 1.4, drawing directly on the evidence presented in Sections 4.2–4.10.

4.11.1 RQ1 Effective Preprocessing for Bangla Text

The combination of Unicode NFC normalisation, systematic noise/artefact removal, explicit outlet-marker stripping, punctuation-aware spacing, and the deliberate avoidance of unreliable stemming (Section 3.4) is shown, through the feature-importance evidence of Section 4.8 and the leakage discussion of Section 4.10, to be effective specifically because it removes shortcut signal while preserving genuine stylometric cues; omitting the outlet-marker and artefact-removal steps in particular would be expected to inflate reported performance without improving genuine detection capability.

4.11.2 RQ2 Optimal Feature Extraction Technique

No single feature family is individually sufficient: the strongest classical models (Section 4.3) combine word-level TF-IDF, character-level TF-IDF, and the hand-crafted meta-feature set, and the feature-importance analysis (Section 4.8) shows that engineered meta-features and n-gram features both contribute materially rather than one dominating the other. The strongest overall system (Section 4.6) further combines this feature-rich classical representation with a transformer's learned contextual representation at the ensembling level, suggesting that representational diversity across feature types, not just across model architectures, contributes to the final result.

4.11.3 RQ3 Deep Learning vs Traditional ML Performance

On this ~10,000-article corpus, classical machine-learning models specifically the gradient-boosting family consistently outperform all five deep-learning architectures trained from scratch (Section 4.3 vs Section 4.4; compare LightGBM's 86.40% F1 with BiLSTM-Attention's 83.97% F1). The training-curve evidence of Figure 4.7 indicates this is attributable to overfitting driven by limited training-set size rather than to any fundamental architectural inadequacy, directly answering RQ3: at this scale, the comparatively modest size of the Bengali fake-news corpus favours classical models over deep architectures trained from scratch.

4.11.4 RQ4 Effectiveness of Bangla-BERT

Fine-tuned BanglaBERT under-performs both fine-tuned XLM-RoBERTa and every classical and ensemble model in this study (Section 4.5), despite being pretrained specifically on Bengali text. This indicates that, contrary to the intuition that a language-specific pretrained model should have an inherent advantage, the benefit of transformer pretraining in this setting is dominated by the limited size of the task-specific fine-tuning corpus (7,191 training articles) rather than by the language-specificity of the pretraining

data directly motivating the future-work direction, discussed in Section 6.6.1 and Section 6.6.2, of fine-tuning on a larger Bengali fake-news corpus.

4.11.5 RQ5 Linguistic Cues for Fake News Identification

Character entropy, lexical diversity, headline–content Jaccard overlap, and journalistic-rigour markers (quotation verbs, named designations, dates, and organisation references) are consistently identified as influential predictors across three independent analyses LightGBM gain-based importance, Logistic Regression coefficients, and SHAP values (Section 4.8) and the corresponding absence of collection-artefact tokens among these top features provides direct evidence that the models' decision boundaries rely on genuine linguistic and stylistic cues, not on incidental artefacts of how the dataset happened to be collected, directly answering RQ5 in the affirmative.

4.12 Visualization of Results

This section consolidates the remaining supporting visualisations referenced throughout Sections 4.3–4.9, grouped by type for ease of reference.

4.12.1 Confusion Matrices

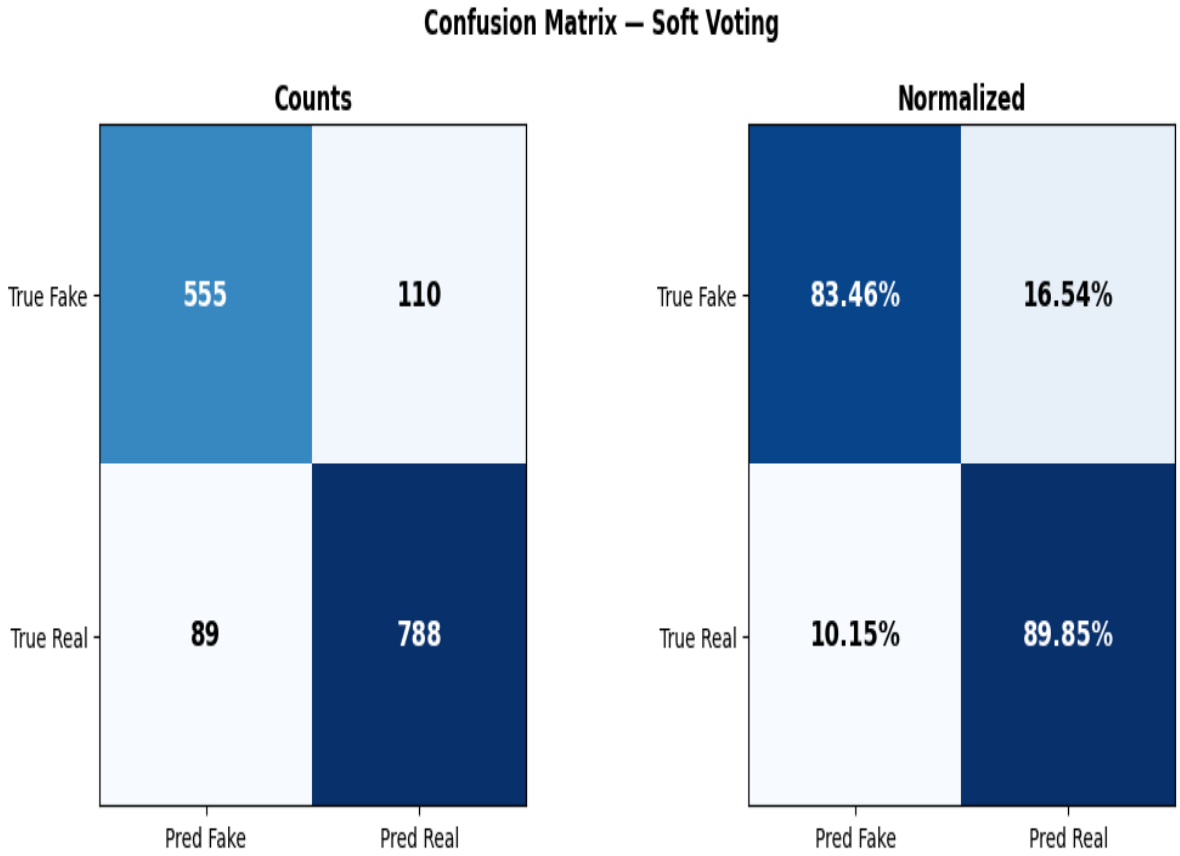


Figure 4.6: Confusion matrix of the best model (Soft Voting) — raw counts and normalized.

Figure 4.6 presents the same confusion matrix twice for complementary readability: the left panel shades each cell by raw test-set count and the right panel shades the identical

cell layout by row-normalised percentage, so that class-conditional error rates can be compared directly across rows despite the underlying class sizes being unequal (665 Fake versus 877 Real, Section 3.10.1). Rows correspond to the ground-truth label and columns to the Soft-Voting ensemble's prediction at its default 0.50 decision threshold, a convention followed consistently throughout this chapter.

Table 4.2 restates the raw confusion-matrix counts underlying Figure 4.6 in tabular form. Of 665 genuinely Fake test articles, 555 (83.5%) are correctly identified; of 877 genuinely Real test articles, 788 (89.9%) are correctly identified.

Table 4.2: Confusion matrix of the best model (Soft-Voting ensemble) on the test set

	Predicted Fake	Predicted Real	Total
Actual Fake	555	110	665
Actual Real	89	788	877
Total	644	898	1,542

By convention, and as followed throughout this thesis, rows of Table 4.2 correspond to the ground-truth label and columns to the model's predicted label, so that the two off-diagonal cells represent, respectively, false negatives and false positives with respect to the Fake class. Expressing these same error counts as the row-normalised percentages already visualised in Figure 4.6 is what allows them to be compared directly with Table 4.3's per-class recall figures below, despite the Fake and Real row totals being of different sizes (665 versus 877).

Table 4.3: Per-class Precision, Recall, and F1-score of the best model

Class	Precision	Recall	F1-Score
Fake	0.862	0.835	0.848
Real	0.878	0.899	0.888

Precision and recall are reported per class in Table 4.3 rather than only as a single averaged figure because, for a task such as fake-news detection, the practical cost of a false negative a fabricated article reaching readers unflagged and a false positive a genuine article wrongly suppressed or flagged are not interchangeable. Presenting both classes' figures side by side allows a deploying organisation to judge whether the model's error profile matches its own operational risk tolerance, rather than relying on a single blended number that could obscure a meaningful asymmetry between the two error types.

The relatively balanced per-class scores in Table 4.3 an F1-score of 0.848 for the Fake class and 0.888 for the Real class confirm that the best model does not simply default toward the majority class, a healthy property given the near-balanced class distribution established in Section 4.2.1.

Table 4.4: Test accuracy by article length band

Article Length Band	Approx. Test Accuracy
Under 100 words	≈ 83%
100–400 words	≈ 86–88%
Over 400 words	≈ 90%

The four length bands in Table 4.4 are intentionally coarse, a deliberate simplification of the continuous relationship already visualised in the right panel of Figure 4.10, so that the length effect on accuracy can be quoted as a small number of interpretable figures rather than only read off a continuous curve, making the finding easier to cite and reason about in the discussion of Section 4.11 and Chapter 5.

4.12.2 ROC Curves

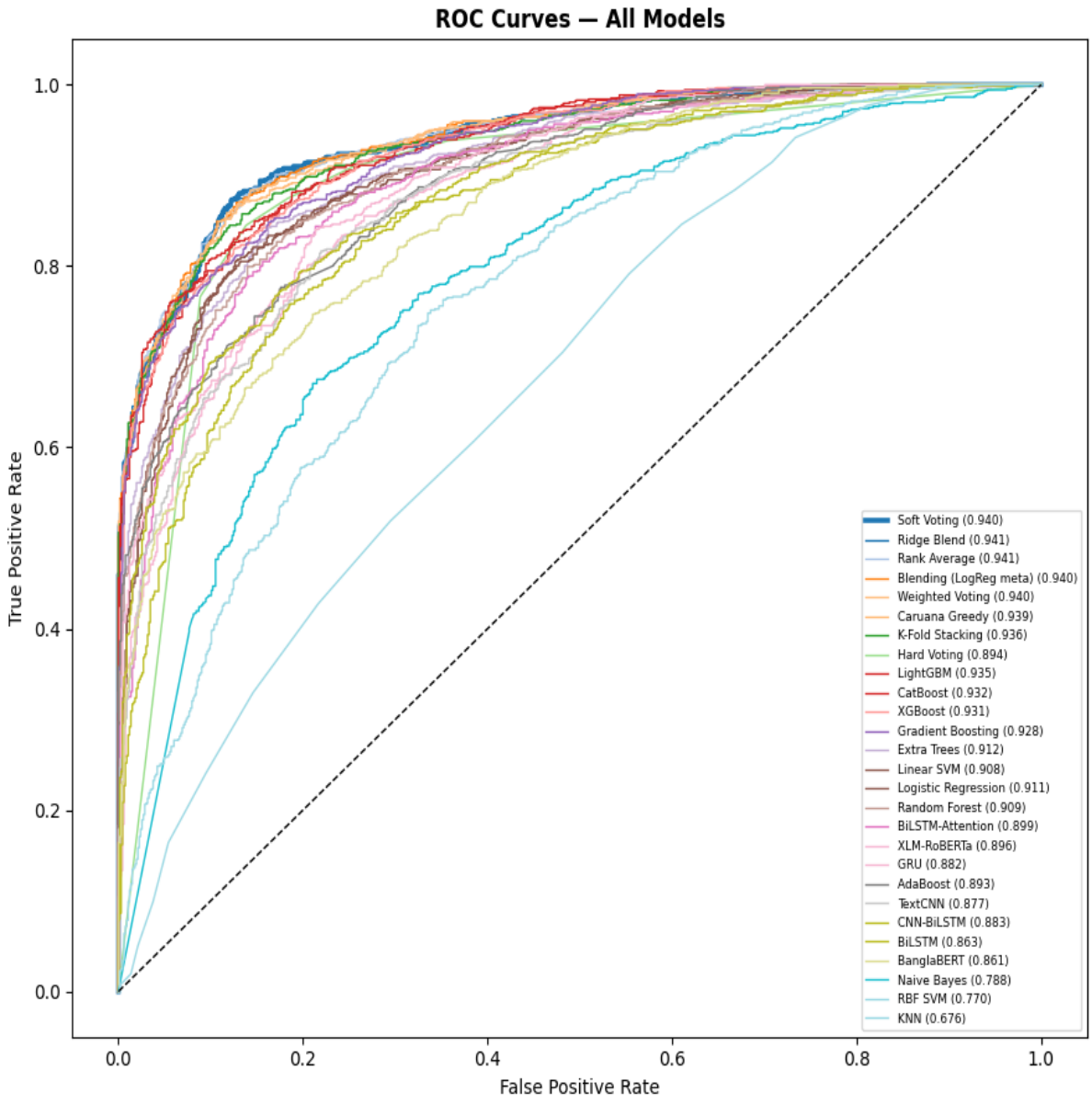


Figure 4.4: Receiver Operating Characteristic (ROC) curves for all models.

Every one of the twenty-seven models' ROC curves is plotted on shared axes, with the legend reporting each model's individual AUC value; the diagonal dashed line marks the theoretical performance of a non-discriminating random classifier, against which every trained model's curve visibly and substantially improves. Because ROC-AUC aggregates performance across all possible decision thresholds simultaneously (Section 3.11.3), this figure allows the full ranking of Table 4.1 to be cross-checked visually without committing to any single operating threshold.

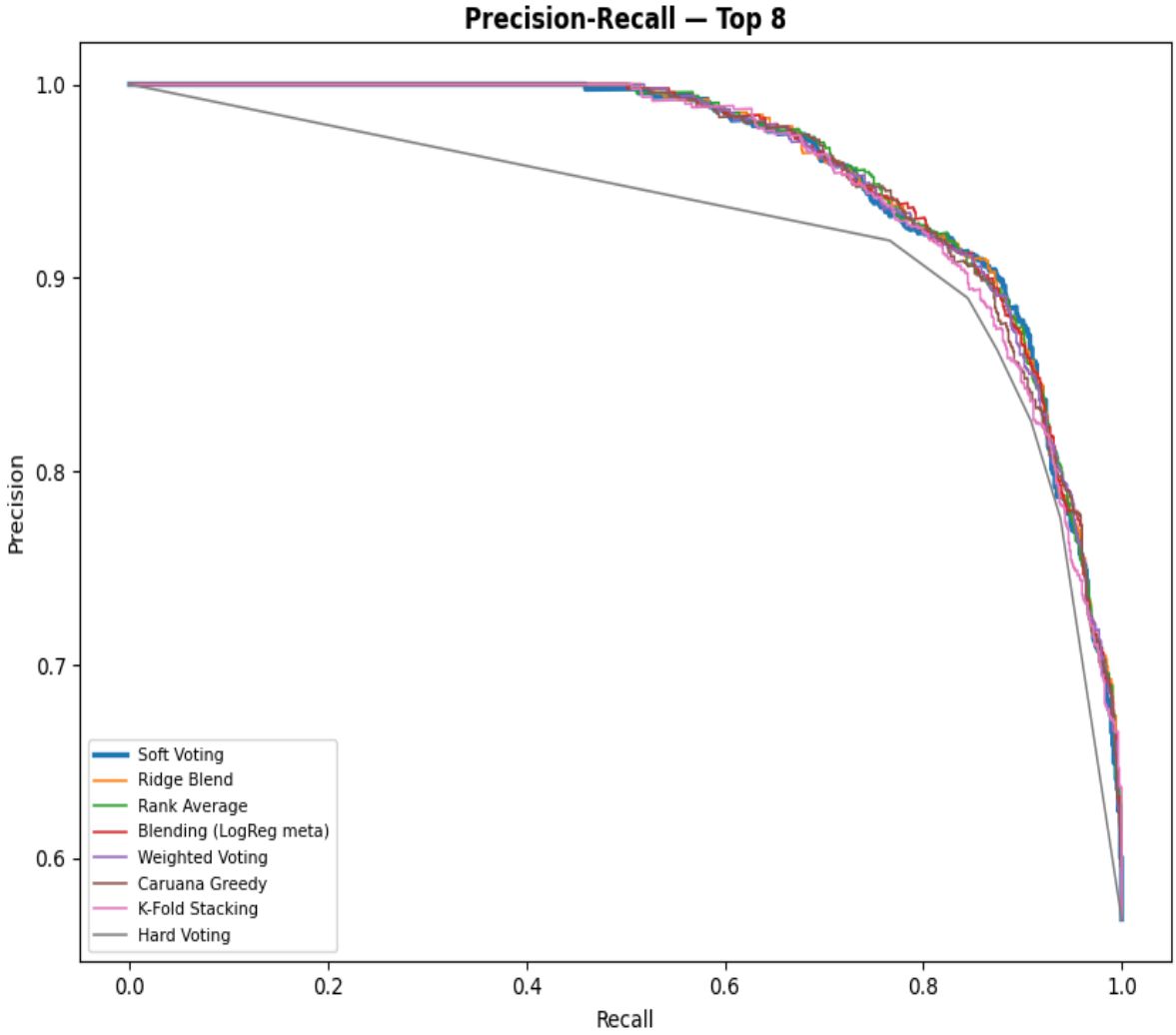


Figure 4.5: Precision–Recall curves for the top-performing models.

Restricting the comparison to the eight top-performing ensemble strategies and switching from ROC to Precision–Recall axes makes the small performance differences among these closely matched strategies easier to distinguish than on a shared ROC plot, where their curves would otherwise overlap almost completely near the top-left corner. Hard Voting is the only member of this group whose curve visibly separates from the other seven, consistent with its comparatively lower F1-score in Table 4.1.

Figure 4.4 confirms, across the full range of decision thresholds, the same ordering established by F1-score in Table 4.1: the ensemble strategies and gradient-boosting models trace curves closest to the top-left corner (highest AUC), while KNN and RBF-SVM trace curves closest to the diagonal line of a non-discriminating classifier. Figure 4.5 focuses this comparison on the eight top-performing models via Precision–Recall curves, which are generally considered a more diagnostic view than ROC when both classes carry practical importance, as is the case here (Section 3.11).

4.12.3 Model Comparison Charts

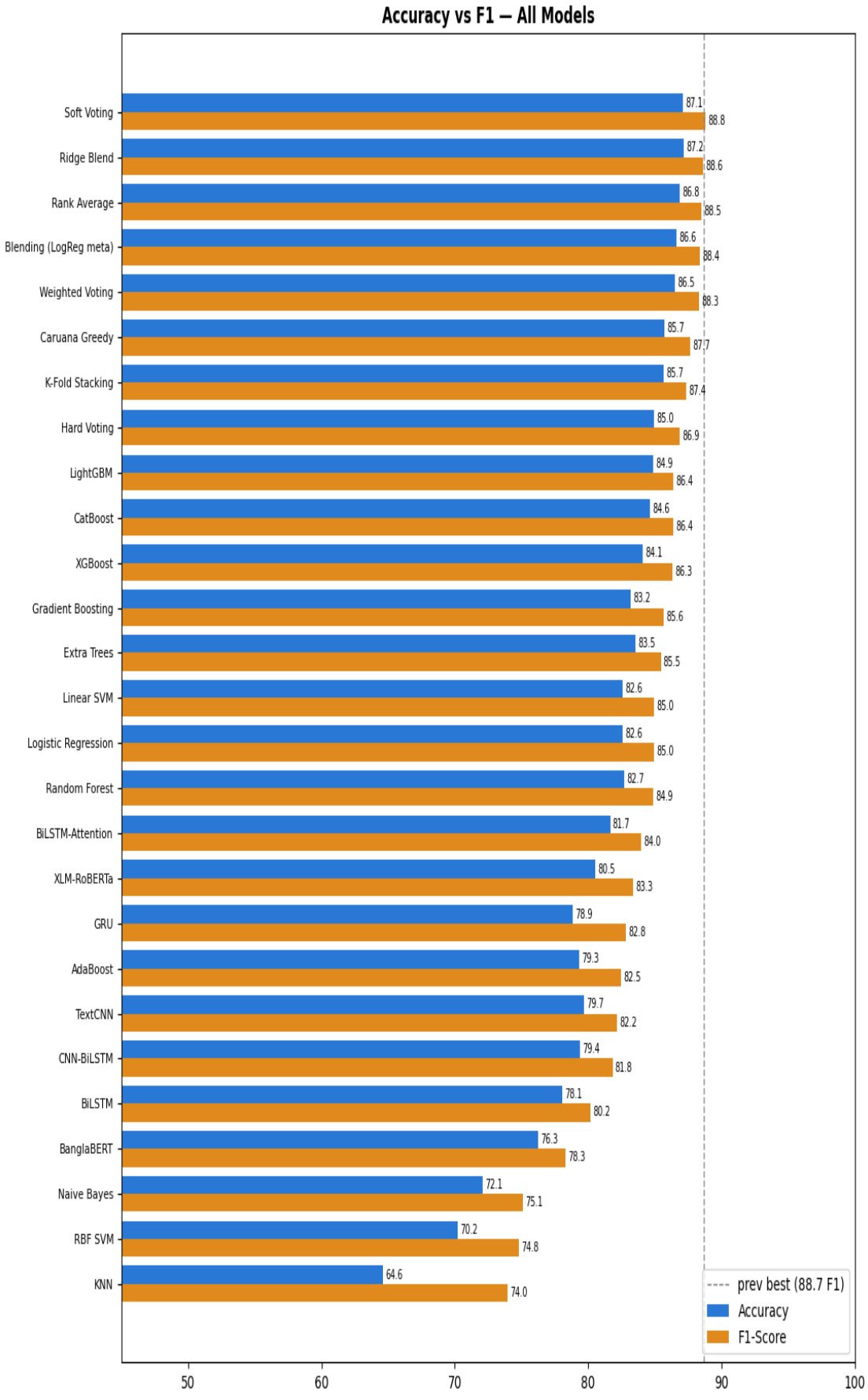


Figure 4.2: Accuracy and F1-score comparison across all twenty-seven models.

Models are ordered along the vertical axis by descending F1-score, with paired horizontal bars reporting accuracy and F1-score for each; a dashed reference line labelled “prev best” in the source figure marks a benchmark F1-score retained from the experimental pipeline, giving the reader a fixed visual anchor against which the final ranking can be judged. The systematic gap visible between the accuracy and F1 bars for lower-ranked models, most pronounced for KNN, reflects the same precision/recall imbalance examined quantitatively in Table 4.1.

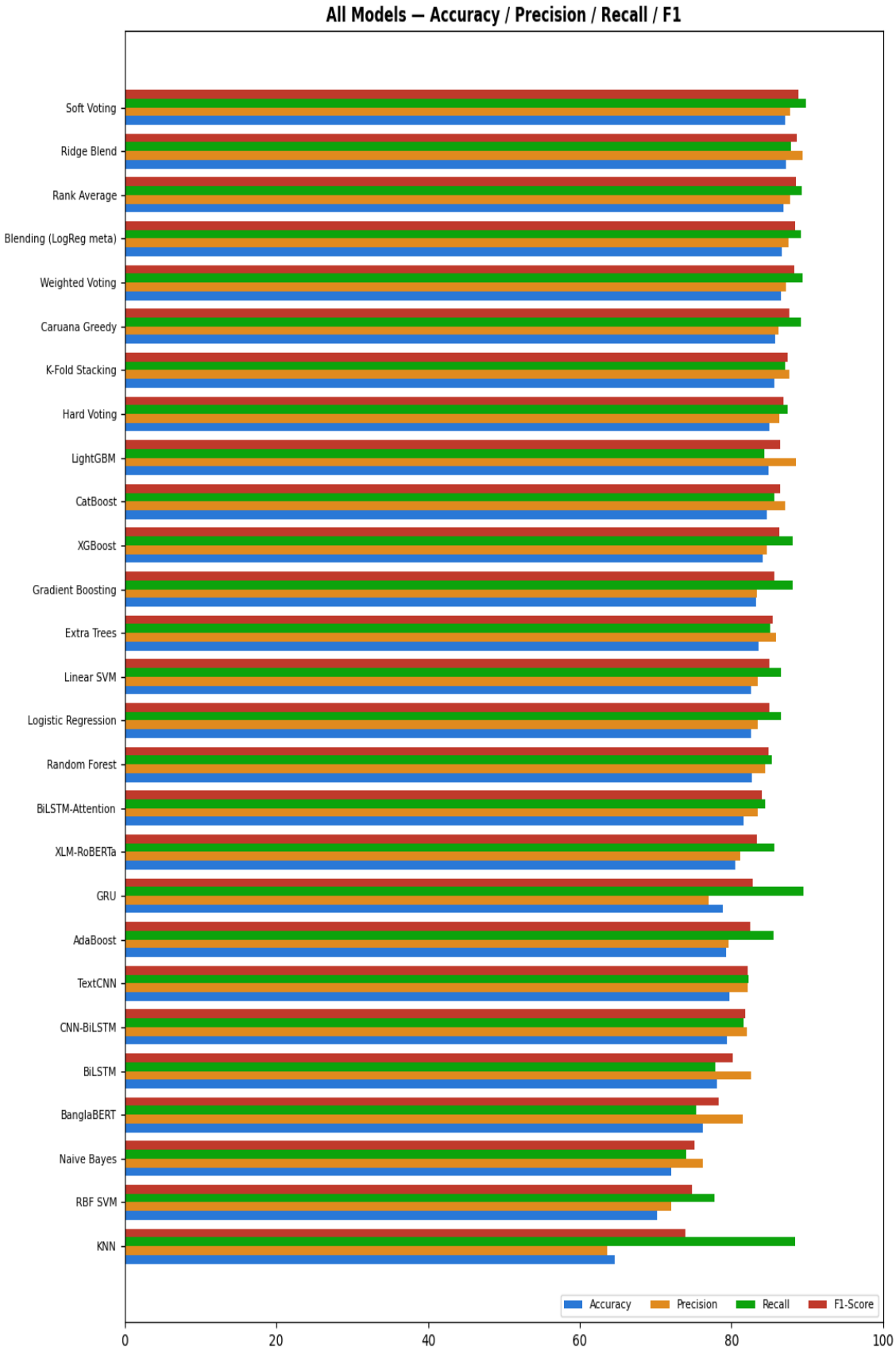


Figure 4.3: Four-metric grouped comparison — Accuracy, Precision, Recall, and F1-score.

Extending Figure 4.2 from two metrics to four, each model is represented by four adjacent bars accuracy, precision, recall, and F1-score enabling within-model comparison across metrics as well as between-model comparison along a single metric. This is what allows a precision/recall imbalance, such as GRU's markedly longer recall bar relative to its precision bar, or KNN's long recall bar paired with a short precision bar, to be identified visually rather than only by cross-referencing the numeric columns of Table 4.1.

Figures 4.2 and 4.3 present the full Table 4.1 result set graphically. Figure 4.3 in particular makes visible the precision/recall imbalance discussed for GRU and KNN in Sections 4.3–4.4 cases where a model's accuracy or F1-score alone would understate a meaningfully skewed error profile.

4.12.4 Feature Importance Plots

The feature-importance and explainability visualisations (Figures 4.9, 4.11, and 4.12) are presented in full in Section 4.8 above, alongside their interpretation, rather than repeated here.

4.12.5 Additional Explainability, Robustness, and Efficiency Visualizations

This subsection presents sixteen further supporting visualisations produced during the experimental work for this thesis, extending the core result figures above with additional diagnostic, robustness, and efficiency evidence: hyperparameter-tuning justification, alternative document-embedding projections, per-feature distributional evidence, threshold and statistical-significance diagnostics, cross-validation stability, and a computational efficiency comparison.

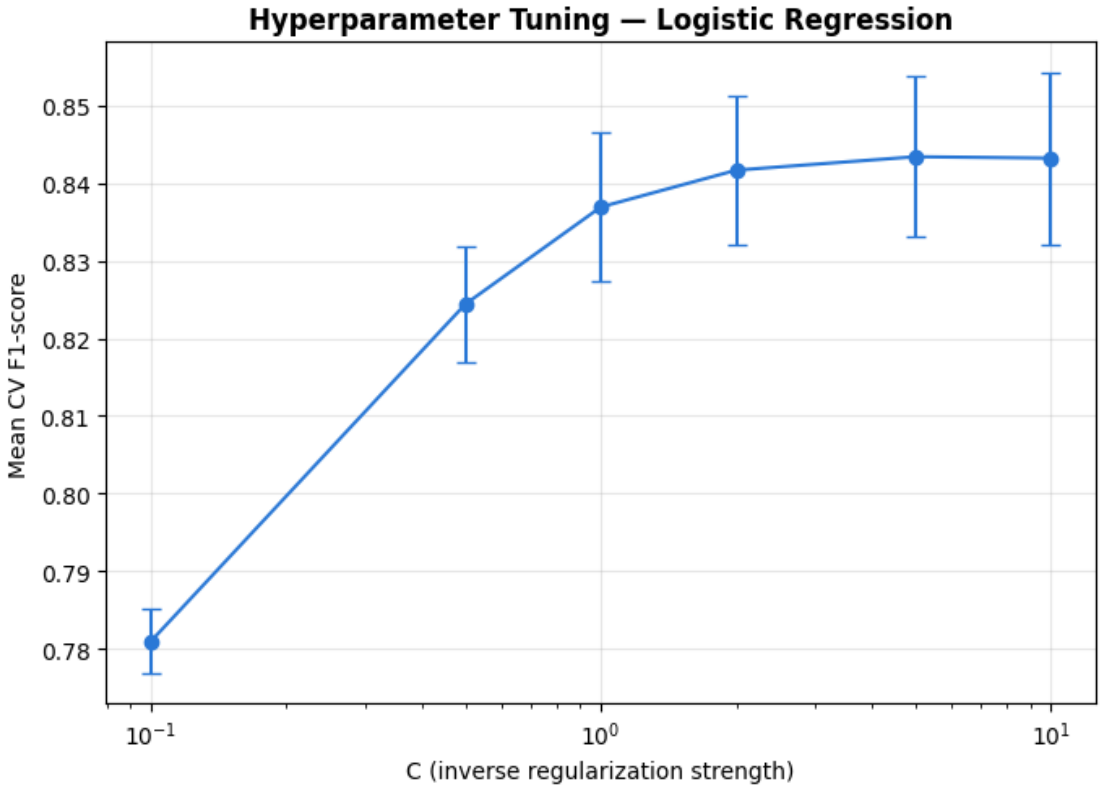


Figure 4.13: Hyperparameter tuning curve for Logistic Regression (mean 5-fold CV F1-score vs. regularisation strength C).

The horizontal axis is plotted on a logarithmic scale to accommodate the wide multiplicative range of the regularisation-strength grid, and the vertical error bars around each point represent variability across the five cross-validation folds rather than a single point estimate. The visibly overlapping error bars from C=2 onward, rather than the mean line alone, are what substantiate the plateau claim made in the accompanying discussion.

Mean CV F1 rises from 0.781 at C = 0.1 to a plateau of 0.843–0.844 for C between 2 and 10, with overlapping error bars at higher C. This confirms the {C: [0.5, 1, 2, 5]} grid actually searched (Section 3.6.1) brackets the plateau rather than under- or over-shooting it.

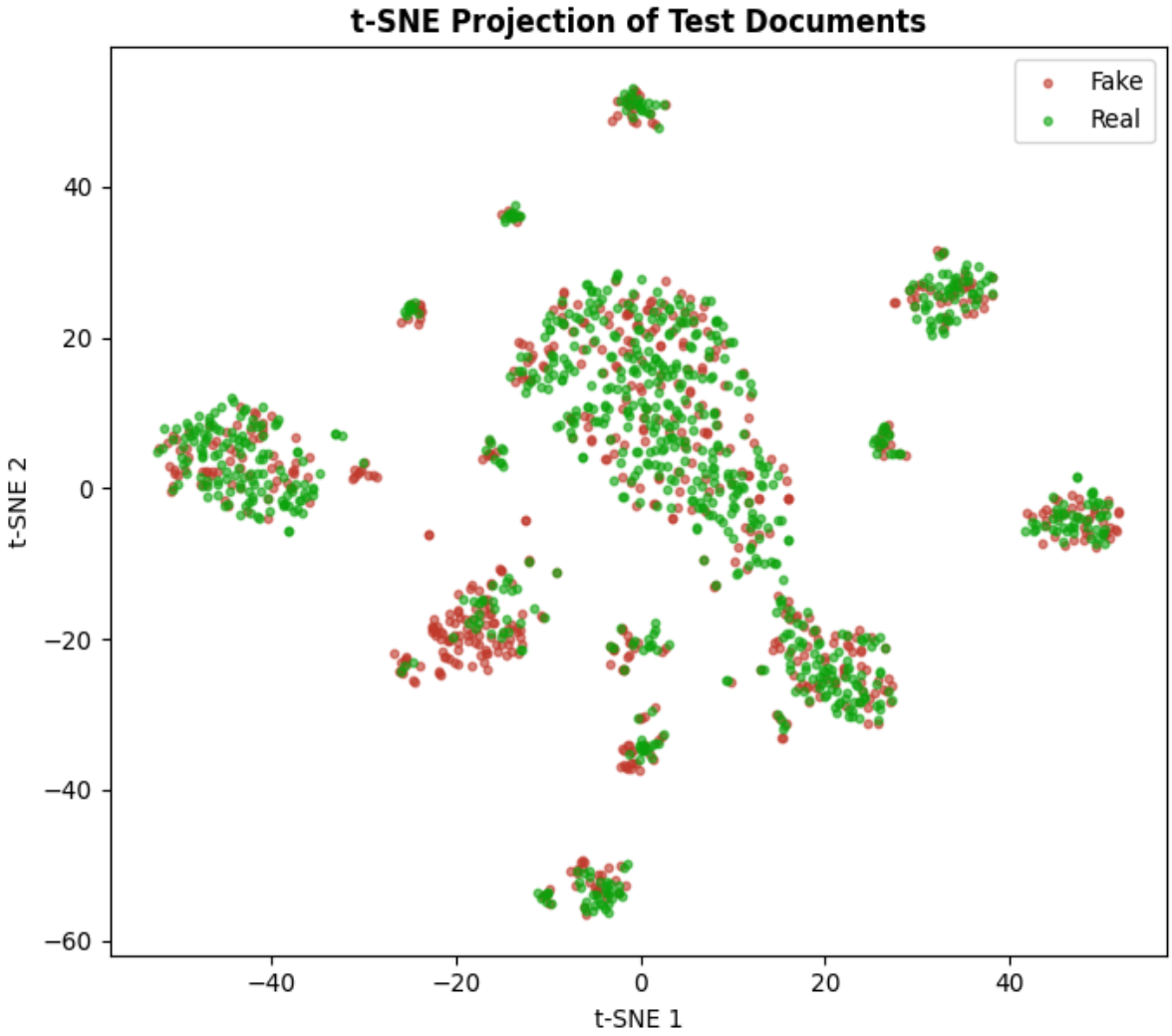


Figure 4.14: t-SNE projection of the 300-dimension SVD document representation for the test partition, coloured by true label.

Each point represents one held-out test article, projected from the 300-dimension SVD document representation (Section 3.6.2) down to two dimensions using t-SNE, a non-

linear technique that preserves local neighbourhood structure at the expense of global distances between clusters. The resulting layout separates into a dozen or so visually distinct clusters, and colouring each point by its true Fake/Real label, rather than by category, is what reveals that these clusters correspond to topic rather than veracity, since both colours appear thoroughly mixed within essentially every cluster.

The projection resolves into distinct clusters corresponding to topical structure rather than the Fake/Real label within every cluster the two classes are thoroughly intermixed. This is consistent with the PCA view (Figure 4.19) and indicates the classifiers' signal comes from stylistic cues rather than an implicit topic detector.

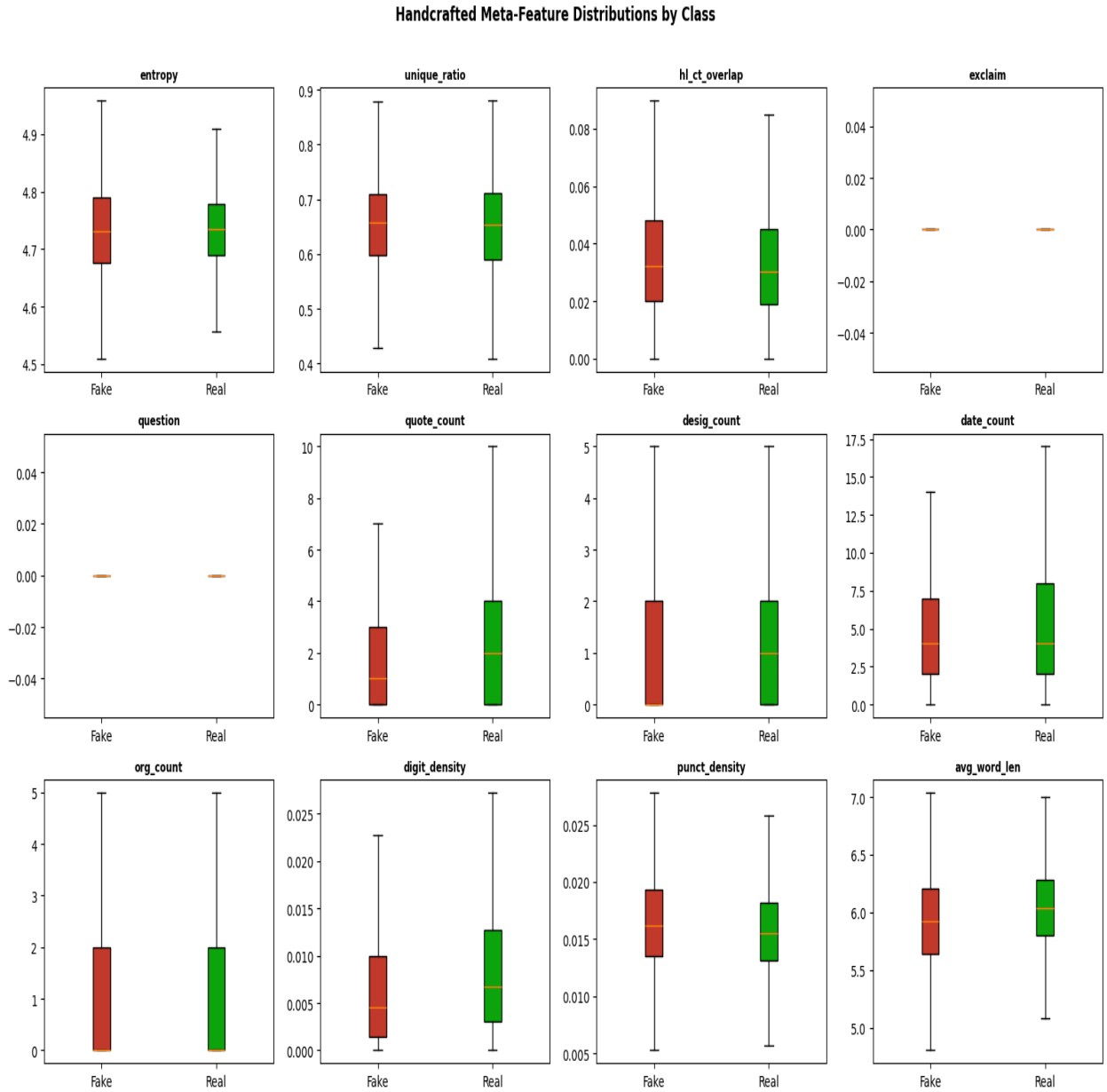


Figure 4.15: Distribution of twelve hand-crafted meta-features by class (Fake vs. Real), test partition.

The twelve hand-crafted meta-features are arranged as a three-by-four grid of paired box plots, one sub-panel per feature, with the Fake-class distribution and Real-class distribution plotted side by side on matching axes within each sub-panel. Presenting all twelve features at a common visual scale per feature allows the reader to judge at a glance which features show a visible median shift between classes and which are nearly indistinguishable between classes, complementing the ranked, aggregate view already provided by Figure 4.11.

Journalistic-rigour features shift toward Real: median quote_count 1 (Fake) vs. 2 (Real), desig_count 0 vs. 1, and higher digit_density consistent with the SHAP findings (Figure 4.11). exclaim/question collapse to near-zero for both classes, showing these two stylistic markers carry little standalone signal on this corpus despite being theoretically motivated (Section 3.5.3.3).

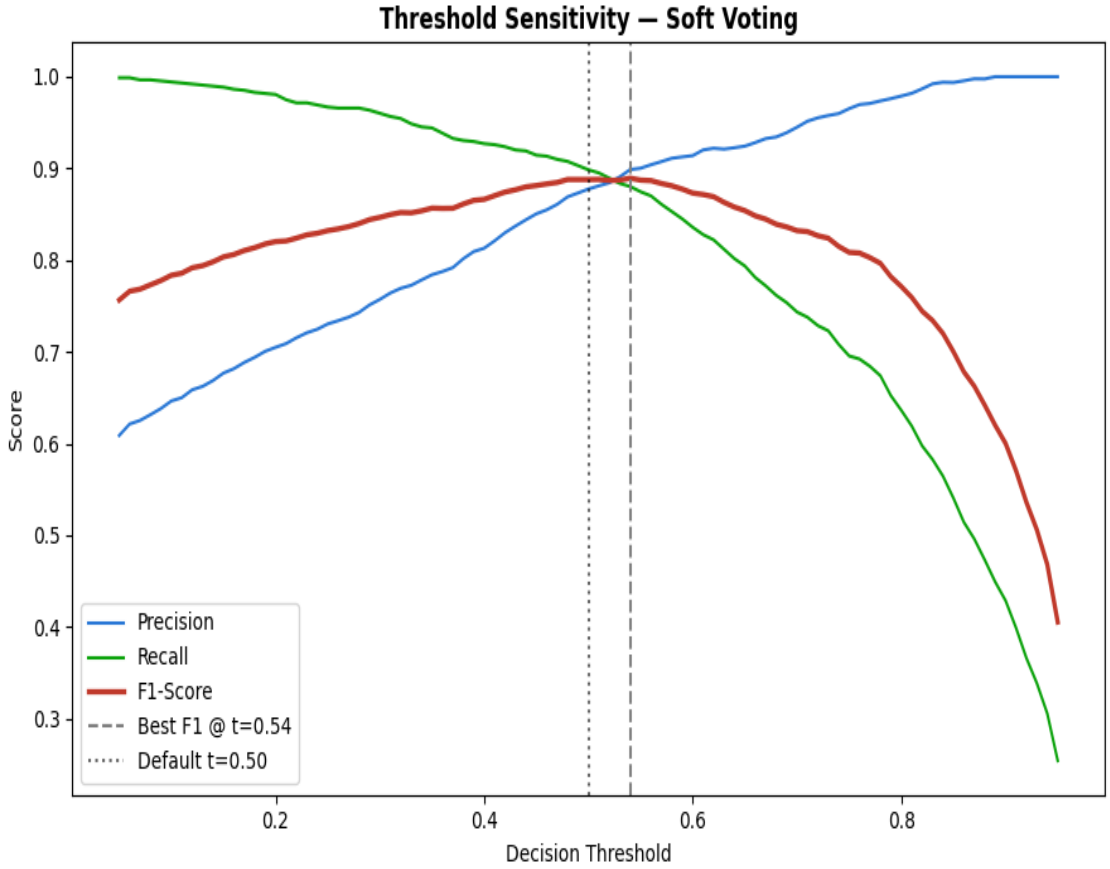


Figure 4.16: Threshold sensitivity curve for the Soft-Voting ensemble: Precision, Recall, and F1-score across the full 0.05–0.95 decision-threshold range.

Precision, Recall, and F1-score are each traced as a continuous function of the decision threshold applied to the Soft-Voting ensemble's predicted probability, swept across the full 0.05–0.95 range. Two vertical reference lines mark the conventional default threshold (0.50) and the empirically best-performing threshold (0.54) identified by this sweep, making visually explicit how close the two are and thereby supporting the decision,

recorded in Section 4.7.2, to retain the simpler default threshold rather than introduce a separately tuned one.

F1-score peaks at threshold 0.54, only marginally above the default 0.50, and the F1 curve is comparatively flat across 0.45–0.60. This visualises why Section 4.7.2 treats the default threshold as a near-optimal operating point for this ensemble.

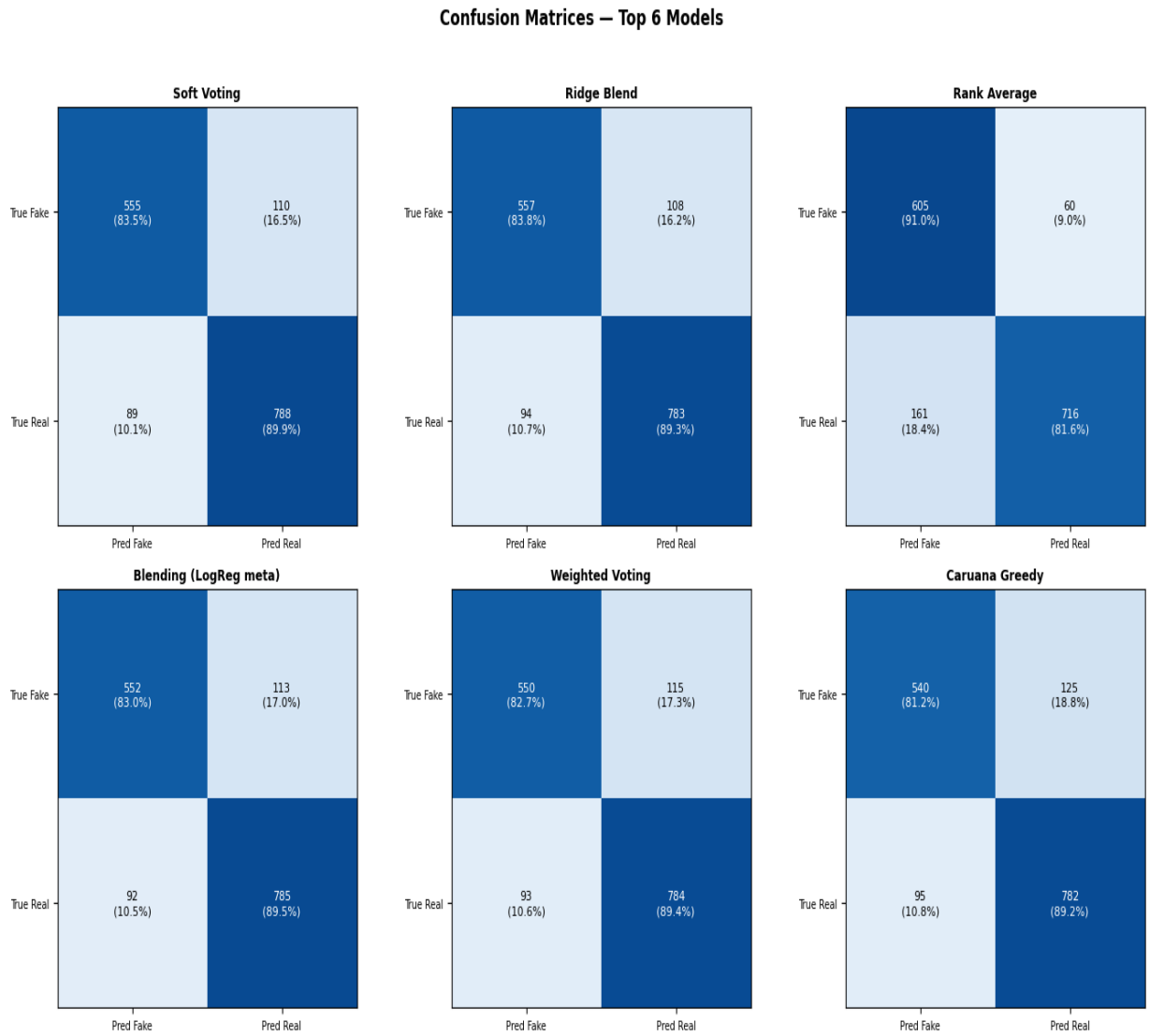


Figure 4.17: Confusion matrices (counts and row-normalised percentages) for the six best-performing models.

Six confusion matrices, one per top-performing ensemble strategy, are arranged in a two-by-three grid using the same counts-plus-row-percentage layout and colour scale as Figure 4.6, so that the error patterns of competing ensemble strategies can be compared cell-by-cell rather than only through their scalar F1-scores. This shared layout is what makes Rank Average's comparatively higher Fake-recall / lower Real-recall trade-off visually identifiable against the other five, more evenly balanced strategies.

The five leading ensembles show similarly balanced error profiles (82–84% Fake recall, 89–90% Real recall). Rank Average is the exception 91.0% Fake recall but only 81.6% Real recall a more Fake-biased trade-off that could suit a fact-checking triage setting preferring to over-flag.

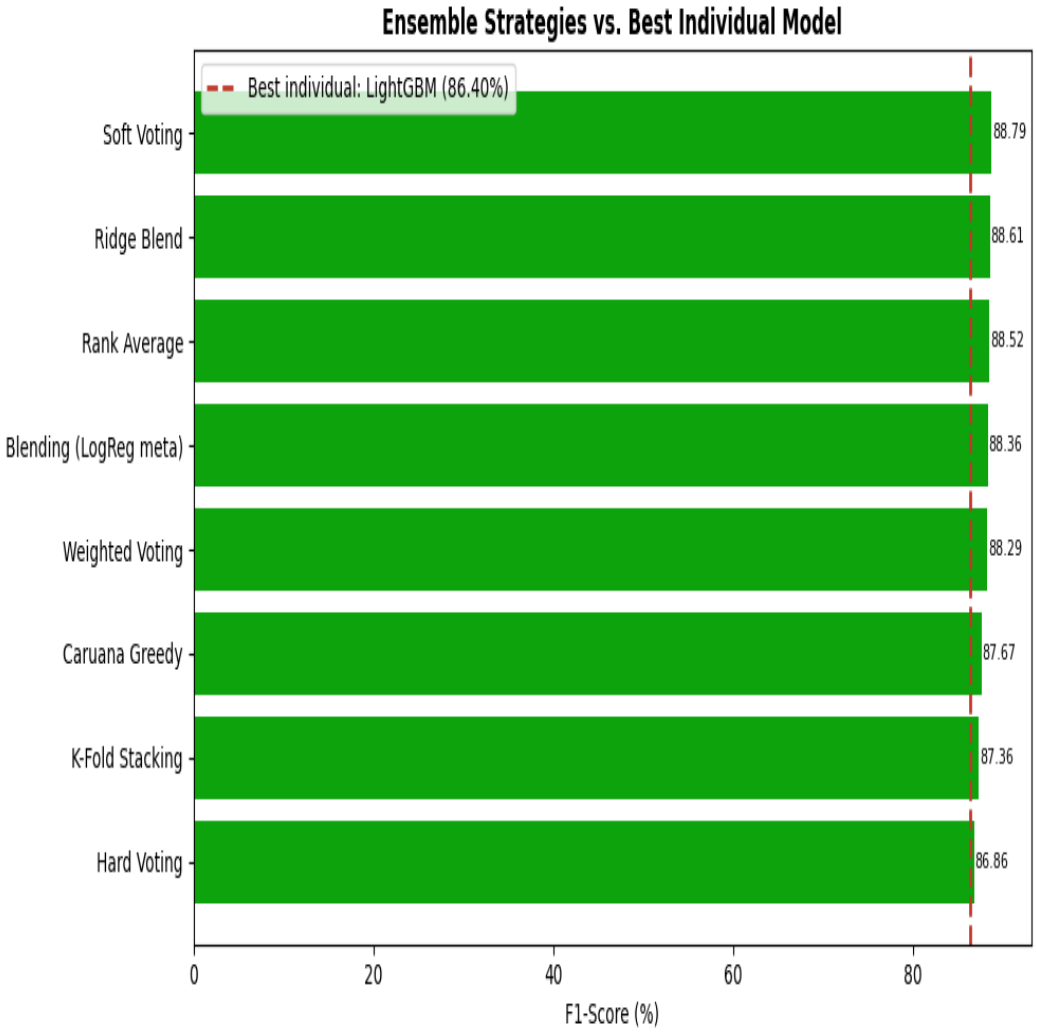


Figure 4.18: F1-score of all eight ensemble strategies compared against the best individual model (LightGBM, 86.40% F1).

All eight ensemble strategies are plotted as horizontal bars on a common F1-score axis against a single dashed vertical reference line marking the strongest individual, non-ensemble model, LightGBM. Because every bar extends beyond this reference line, the figure provides an immediate, single-glance visual confirmation of the ensembling benefit argued in Section 4.6, independent of the more detailed numeric comparison in Table 4.1.

Every ensemble strategy exceeds LightGBM's 86.40% F1 reference line, from Hard Voting (86.86%) to Soft Voting (88.79%). That all eight independently constructed rules land above the same line is evidence of genuine complementary signal among the selected base models (Section 3.9), not one favourably-tuned combination overfitting validation data.

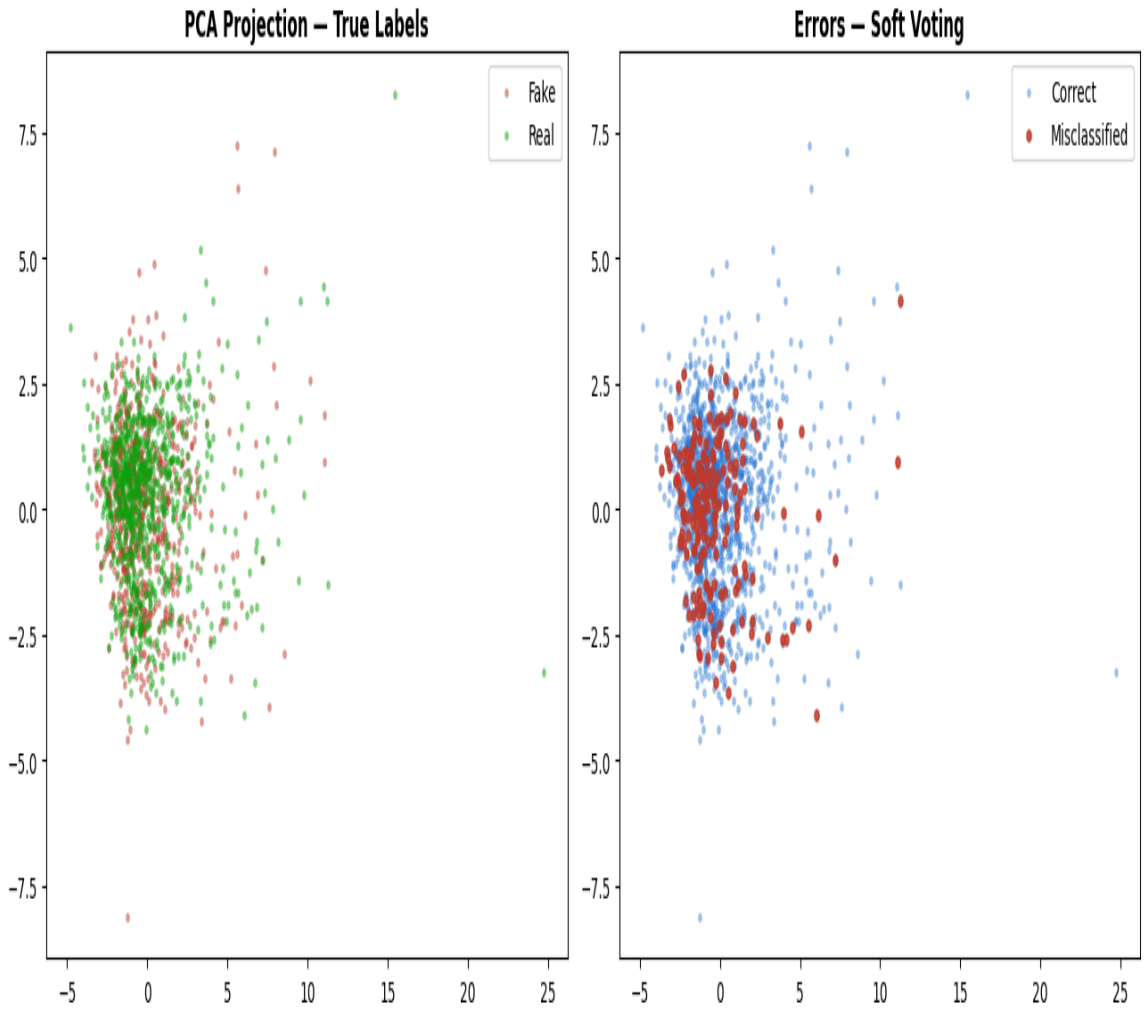


Figure 4.19: PCA projection of the 300-dimension document representation: true labels (left) and Soft-Voting correctness (right).

The same two-dimensional PCA projection of the document representation is plotted twice, coloured first by true class label (left) and then by whether the Soft-Voting ensemble's prediction was correct (right). Using identical point coordinates across both panels means that any region of the left panel where Fake and Real points overlap most heavily can be checked directly, panel-to-panel, against the corresponding concentration of misclassified points on the right, which is the visual basis for the error-clustering claim in the accompanying discussion.

Fake and Real articles are thoroughly intermixed near the origin (left). Misclassified articles (right) concentrate almost entirely within this dense central overlap region and are largely absent from sparser outlying areas consistent with the error-analysis finding (Section 4.9) that errors cluster near the decision boundary.

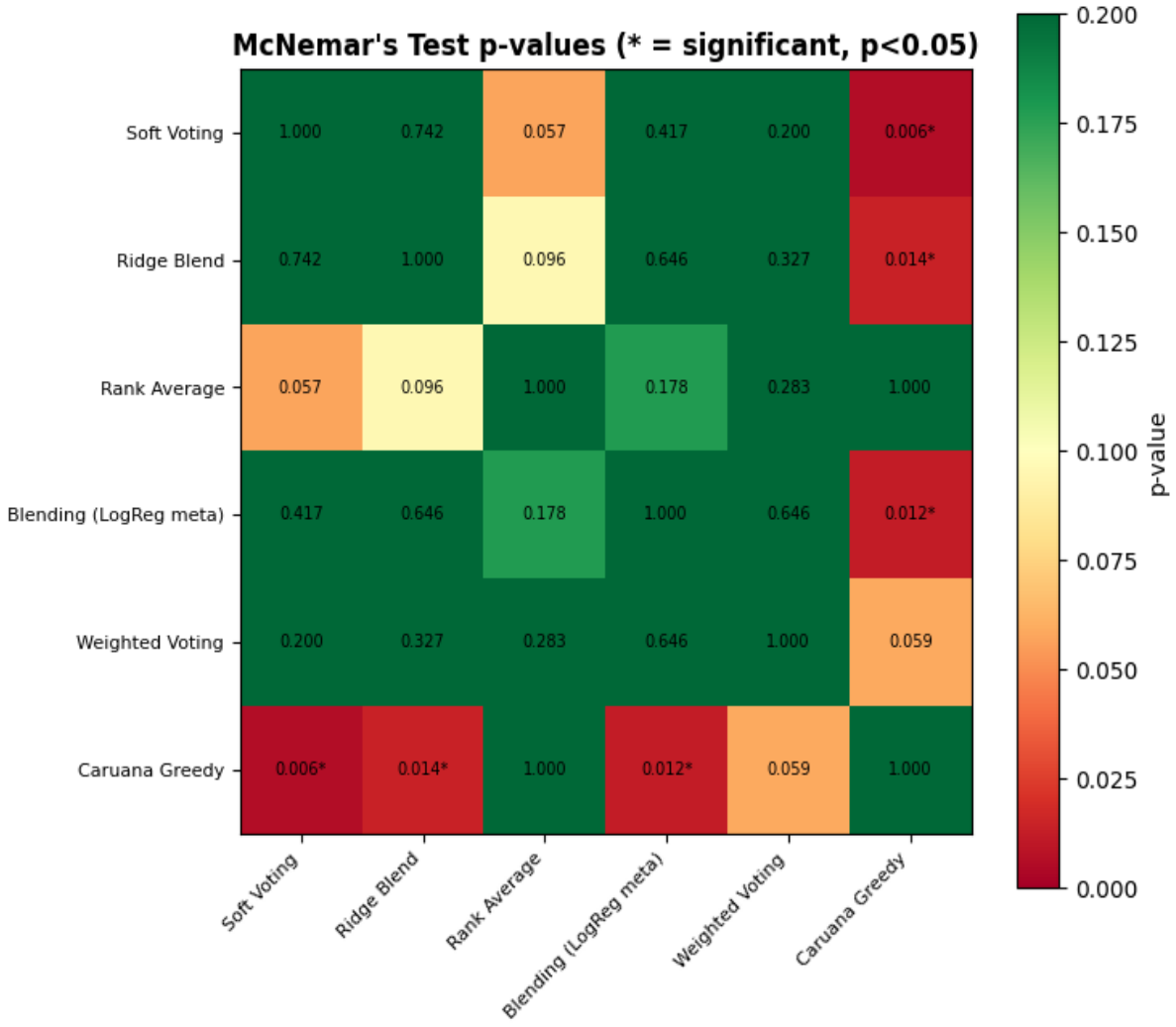


Figure 4.20: McNemar's test p-values for pairwise prediction agreement among the top six models (* = significant, $p < 0.05$).

The heat-map reports pairwise McNemar's-test p-values for all fifteen combinations of the top six models, with the diagonal fixed at 1.000 by construction and a diverging colour scale in which red cells mark statistically significant disagreement ($p<0.05$, additionally flagged with an asterisk) and green cells mark pairs whose error patterns are not distinguishable at conventional significance levels. McNemar's test is used here rather than a simple difference in F1-score because it operates directly on the paired per-article agreement/disagreement pattern between two classifiers, making it a formally appropriate significance test for models evaluated on the same held-out set.

Only four of fifteen pairs are significant ($p < 0.05$), all involving Caruana Greedy versus Soft Voting, Ridge Blend, and Blending. Soft Voting and Ridge Blend the top two strategies are not significantly different ($p = 0.742$), statistically supporting the claim (Section 4.7.2) that the top five ensembles form a robust, near-equivalent performance region.

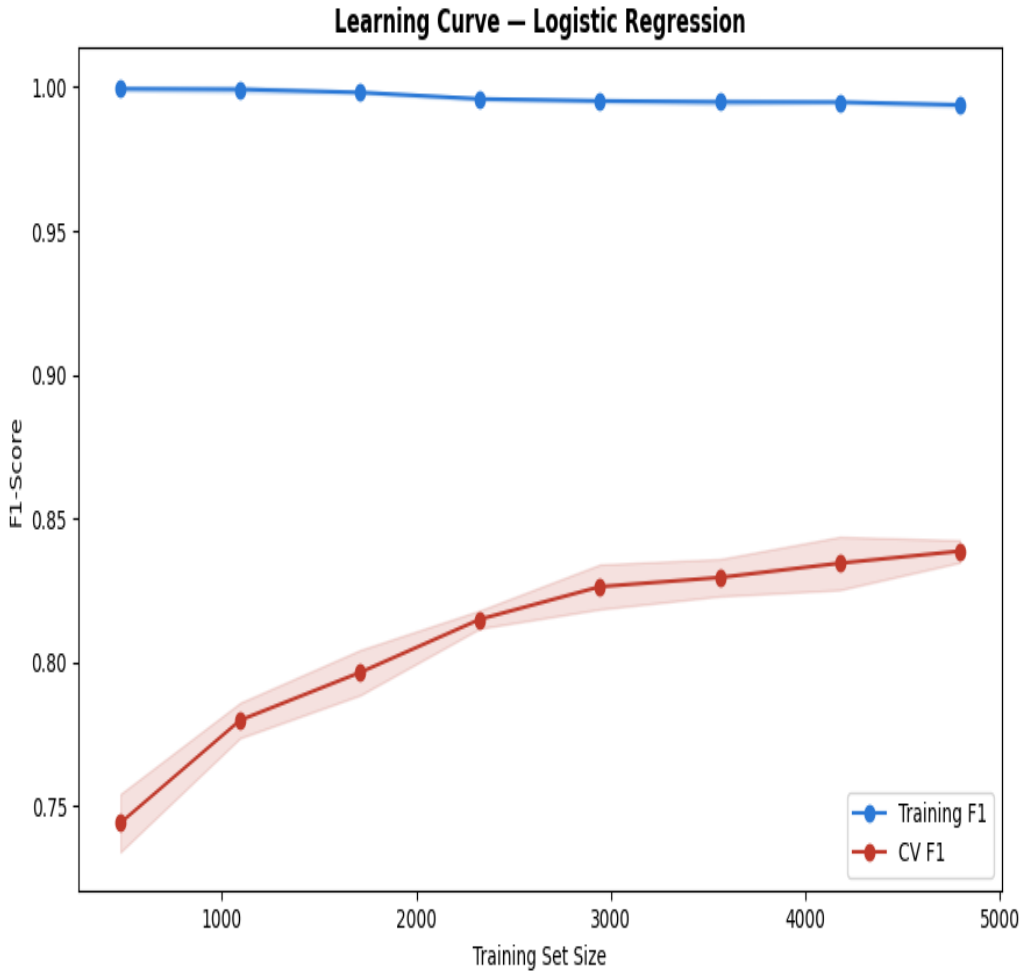


Figure 4.21: Learning curve for Logistic Regression: training and 3-fold cross-validation F1-score vs. training-set size.

Training-set size is increased incrementally along the horizontal axis while both the training-set F1-score and the three-fold cross-validation F1-score are recorded at each size, with a shaded band around the cross-validation curve indicating variability across folds. The persistent, non-narrowing gap between the near-saturated training curve and the still-rising validation curve is the classic diagnostic signature of a model that would benefit from additional training data rather than from further algorithmic tuning, directly supporting the data-scale argument developed in Section 4.11.3.

Training F1 stays saturated near 1.0 throughout, while CV F1 rises steadily from 0.745 (500 examples) to 0.839 (full training size) without plateauing. This complements the per-epoch curves of Figure 4.7 in supporting Section 4.11.3's conclusion that performance is constrained more by training-data volume than by model architecture.

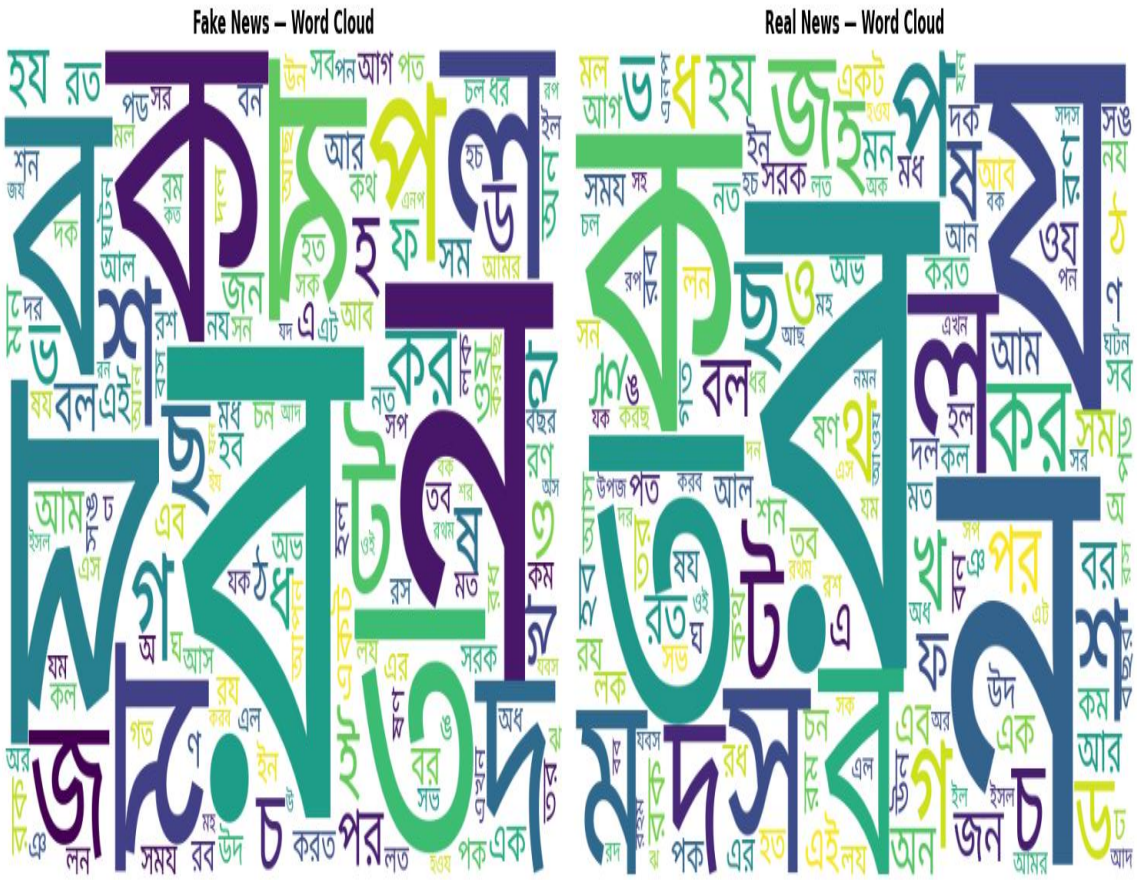


Figure 4.22: Word clouds of the most frequent terms in cleaned article content, Fake (left) and Real (right).

Word size in each cloud is scaled by frequency within the corresponding class's cleaned article content, giving a rapid, qualitative impression of vocabulary composition that complements the quantitative n-gram and Zipf's-law analyses elsewhere in this chapter (Figure 4.9, Figure 4.28). Because word clouds are inherently a qualitative rather than statistically precise visualisation, their role here is corroborative confirming the absence of outlet-marker or scraping-artefact tokens noted quantitatively in Section 4.8 rather than a primary source of evidence in its own right.

Both clouds are dominated by ordinary topical and functional Bengali vocabulary, with no residual outlet name or scraping artefact of the kind identified in Section 3.3.4 a further, visual line of evidence that the cleaning pipeline removed the shortcut signal it targeted.

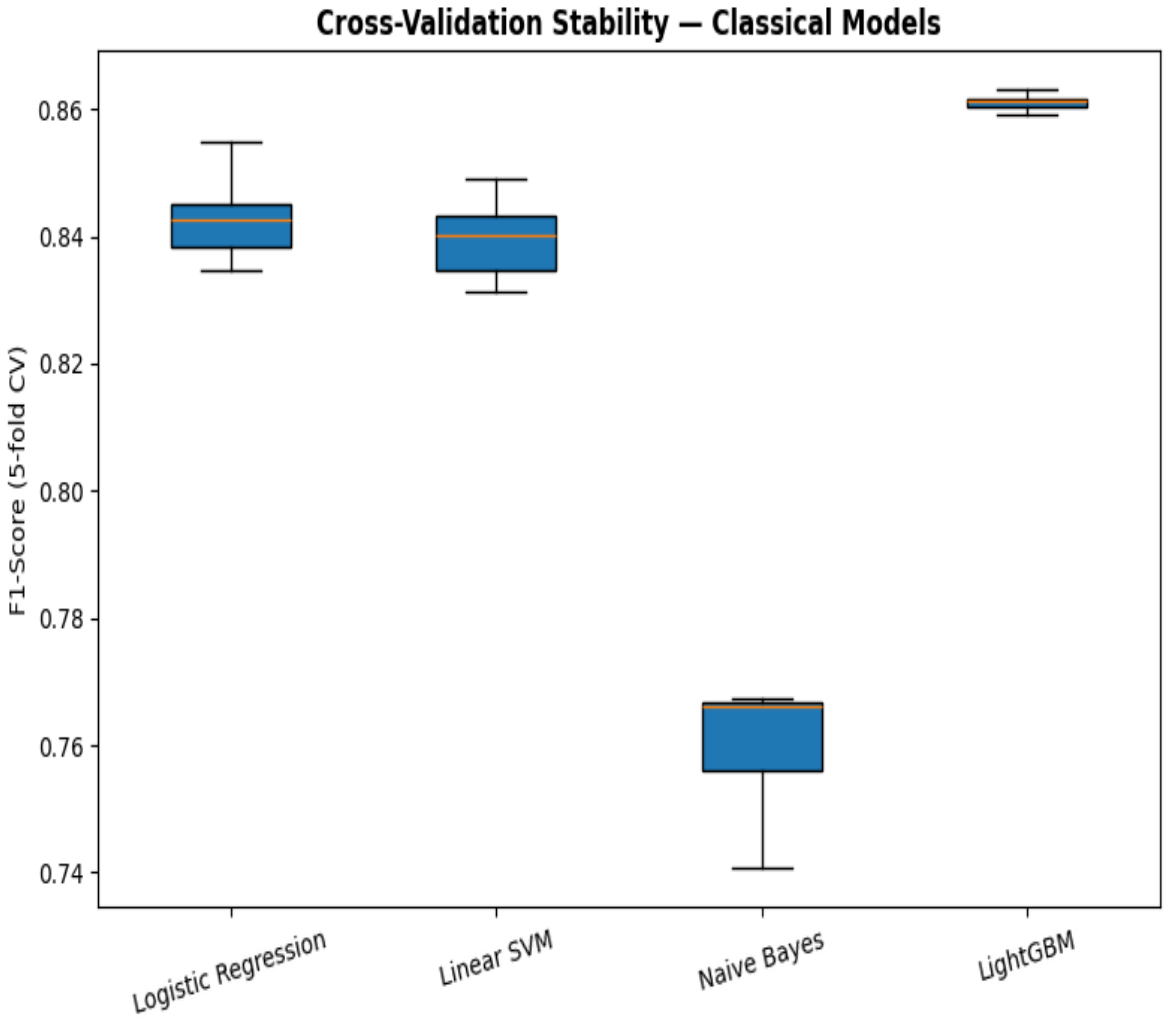


Figure 4.23: Five-fold cross-validation F1-score stability for four classical models with reusable feature matrices.

Five-fold cross-validation F1-scores are summarised as box plots for four classical models chosen specifically because their feature matrices could be cached and reused across folds without prohibitive recomputation cost. Box height, the interquartile range, is the primary quantity of interest here, since a narrow box indicates that a model's performance is a stable property of the data rather than a favourable artefact of one particular fold assignment, as is visibly the case for LightGBM relative to the wider boxes of Logistic Regression and Linear SVM.

LightGBM is both the strongest and most stable (86.11% ± 0.13); Logistic Regression (84.31% ± 0.69) and Linear SVM (83.98% ± 0.63) show moderate spread; Naive Bayes is weakest and least stable (75.93% ± 1.02). LightGBM's narrow spread indicates its advantage is a consistent property of the data, not a lucky split.

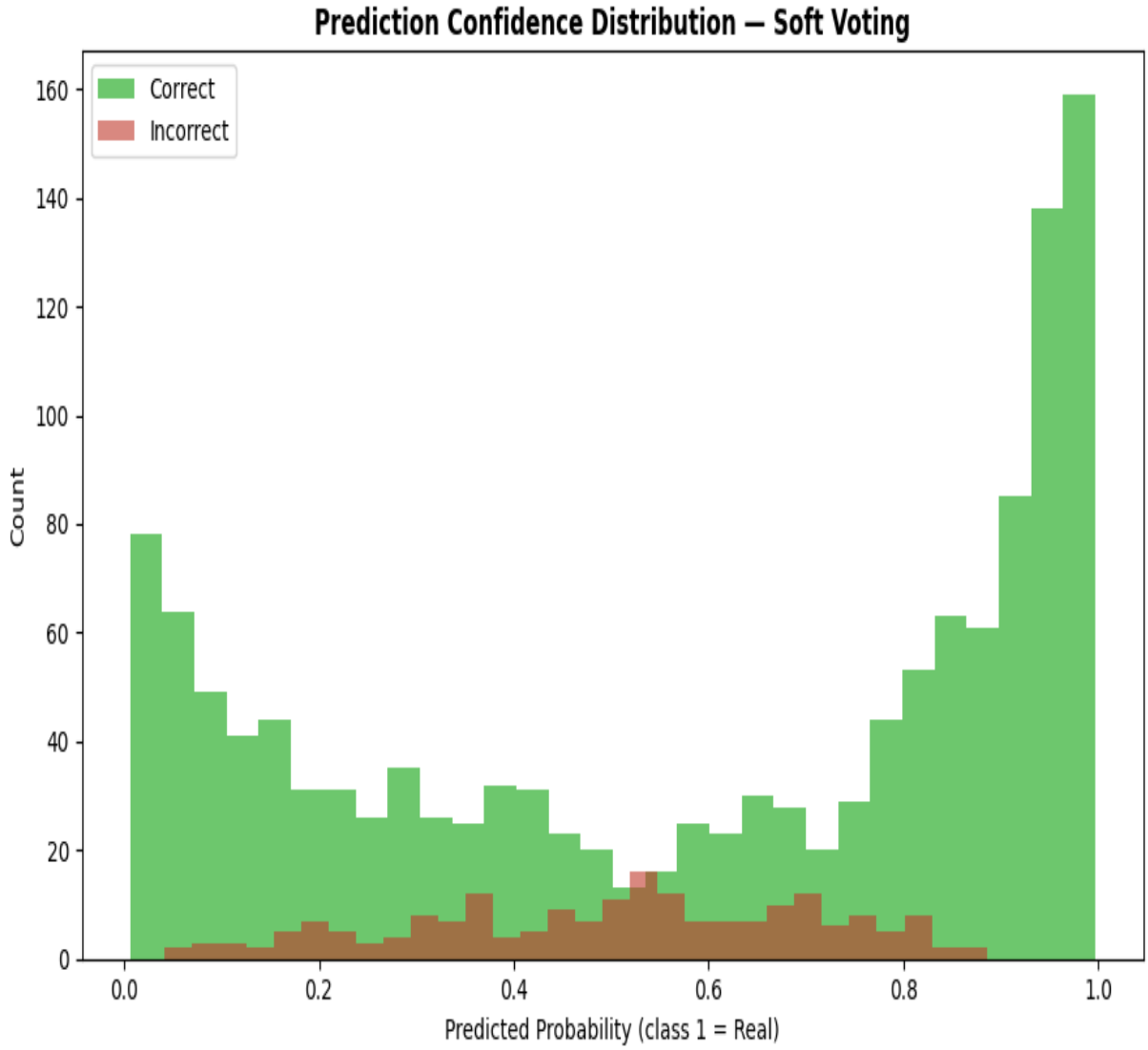


Figure 4.24: Distribution of Soft-Voting predicted probability for correct vs. incorrect test-set predictions.

Predicted probabilities for the Real class are binned into a histogram and split by whether the Soft-Voting ensemble's resulting prediction was correct or incorrect, across the full 0–1 probability range. The pronounced U-shape of the correct-prediction histogram versus the comparatively flat, centrally concentrated incorrect-prediction histogram is a direct visual test of model calibration, since a well-calibrated classifier is expected to reserve its most confident, near-0 or near-1, predictions disproportionately for the cases it gets right.

Correct predictions are strongly bimodal near 0 and 1; incorrect predictions cluster near the 0.4–0.6 decision boundary. This is a favourable calibration property (Section 5.3.3): the model is more uncertain precisely when it is more likely to be wrong, rather than confidently wrong.

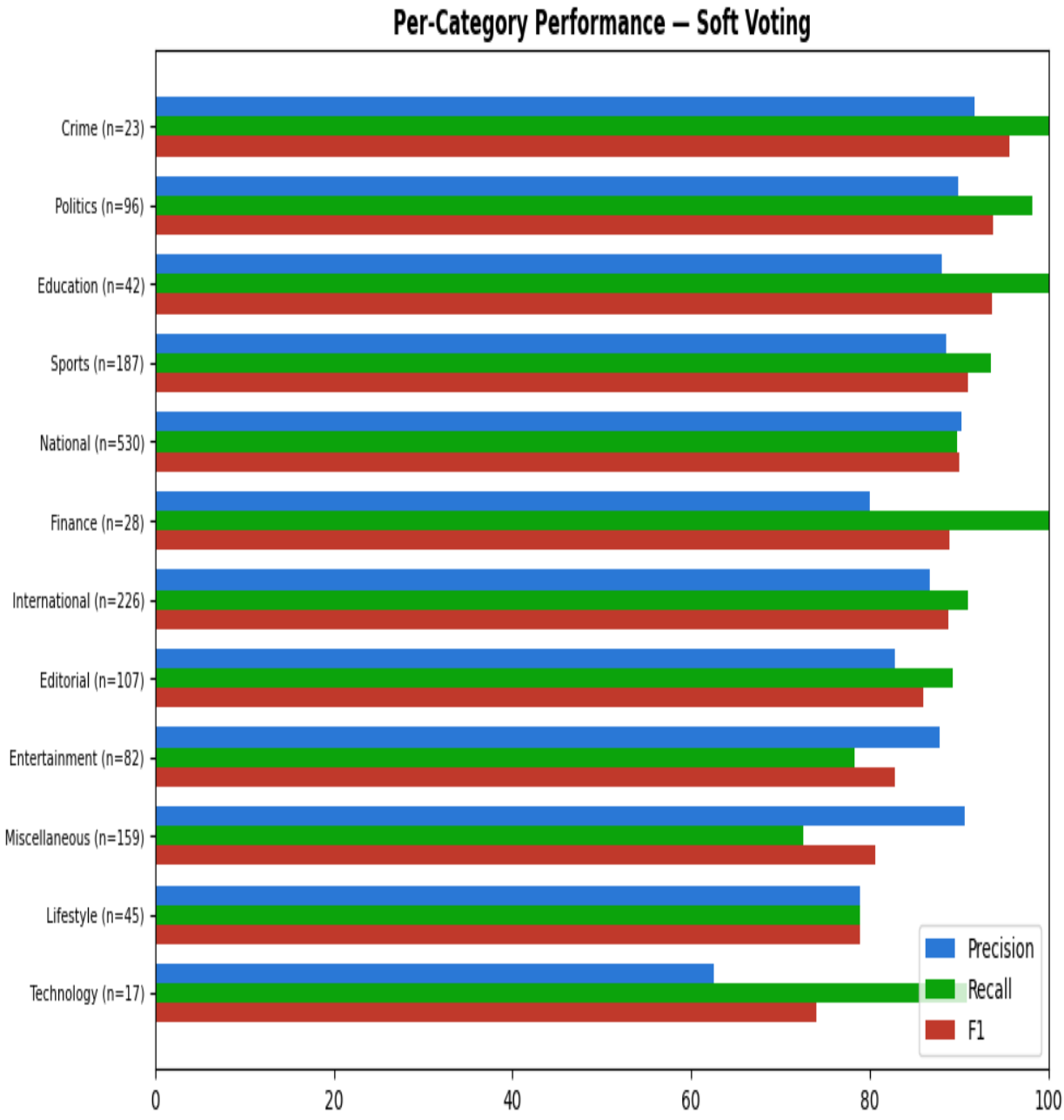


Figure 4.25: Per-category Precision, Recall, and F1-score of the Soft-Voting ensemble on the test partition.

Each of the twelve categories is annotated directly on the vertical axis with its test-set sample size, ranging from n=17 for Technology to n=530 for National, which is essential context for interpreting the adjacent bars: a category's precision, recall, and F1-score are each computed from a correspondingly small or large number of test articles, so apparent differences between the best- and worst-performing categories partly reflect differing estimation reliability and not solely differing genuine model performance, a distinction the raw percentages alone would not convey.

Crime, Politics, and Education exceed 88% F1. Technology (n = 17, the smallest category) has the weakest profile, with precision around 62%, reflecting its training-data

scarcity (Section 3.3.2). Miscellaneous shows the inverse pattern high precision, lower recall illustrating the category-level fairness concern raised in Section 5.3.2.

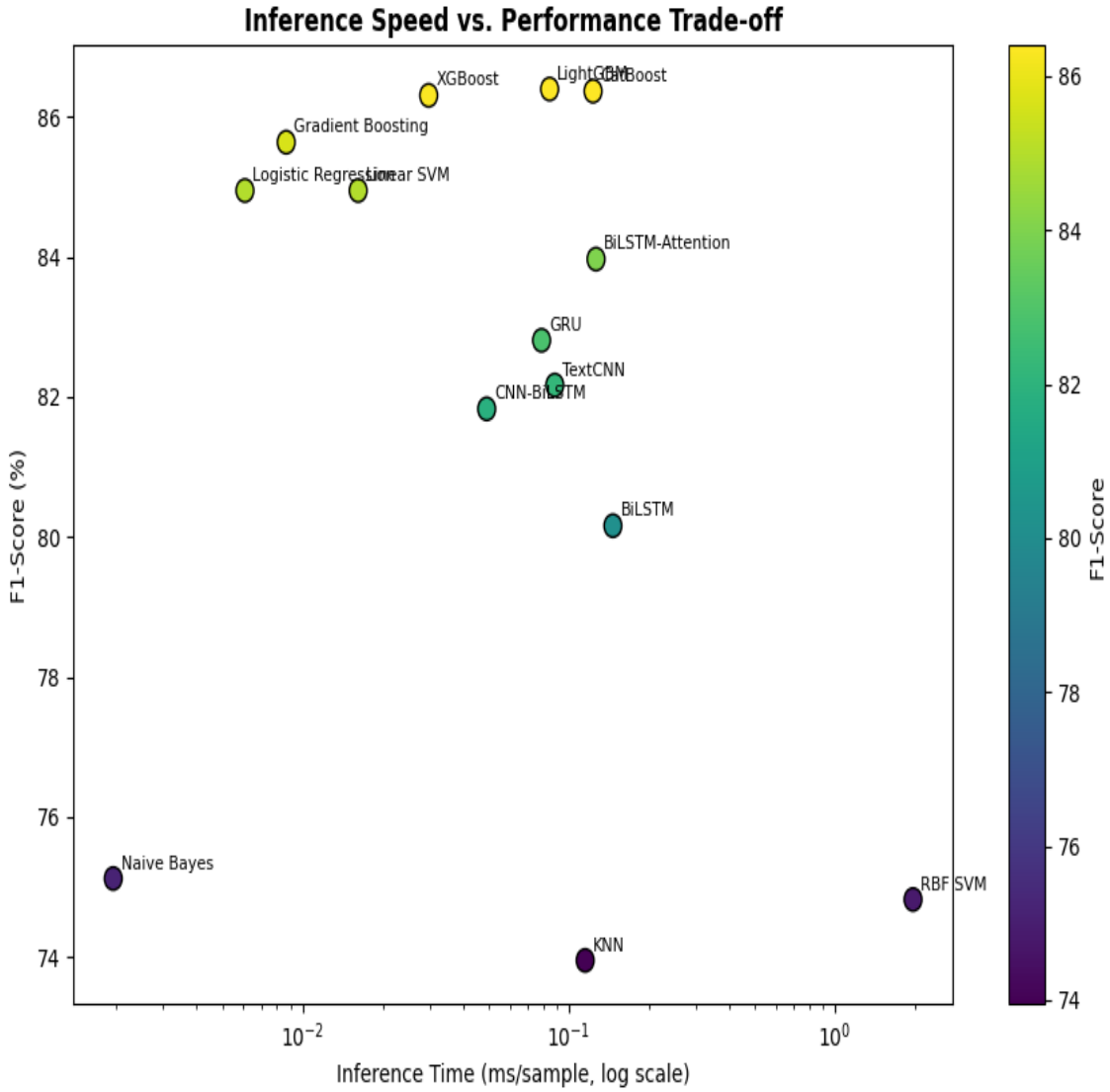


Figure 4.26: Per-sample inference latency versus F1-score trade-off across all classical and deep-learning models still resident in memory.

Each point represents one model plotted by its per-sample inference latency, logarithmic horizontal axis, in milliseconds, against its test-set F1-score, with point colour providing a redundant, continuous encoding of the same F1-score for ease of visual scanning. Because the horizontal axis spans roughly three orders of magnitude, the log scale is necessary to keep the sub-millisecond classical models and the comparatively slower deep-learning models legible on the same plot, directly visualising the accuracy-to-computational-cost trade-off referenced in Section 5.2.4.

The gradient-boosting and linear models combine sub-0.1ms inference with F1 at or above 85%; RBF-SVM and KNN are simultaneously the slowest and weakest. This substantiates the deployment guidance of Section 5.2.4: gradient-boosting models offer

far better throughput than deep architectures for a small F1 sacrifice relative to the full ensemble.

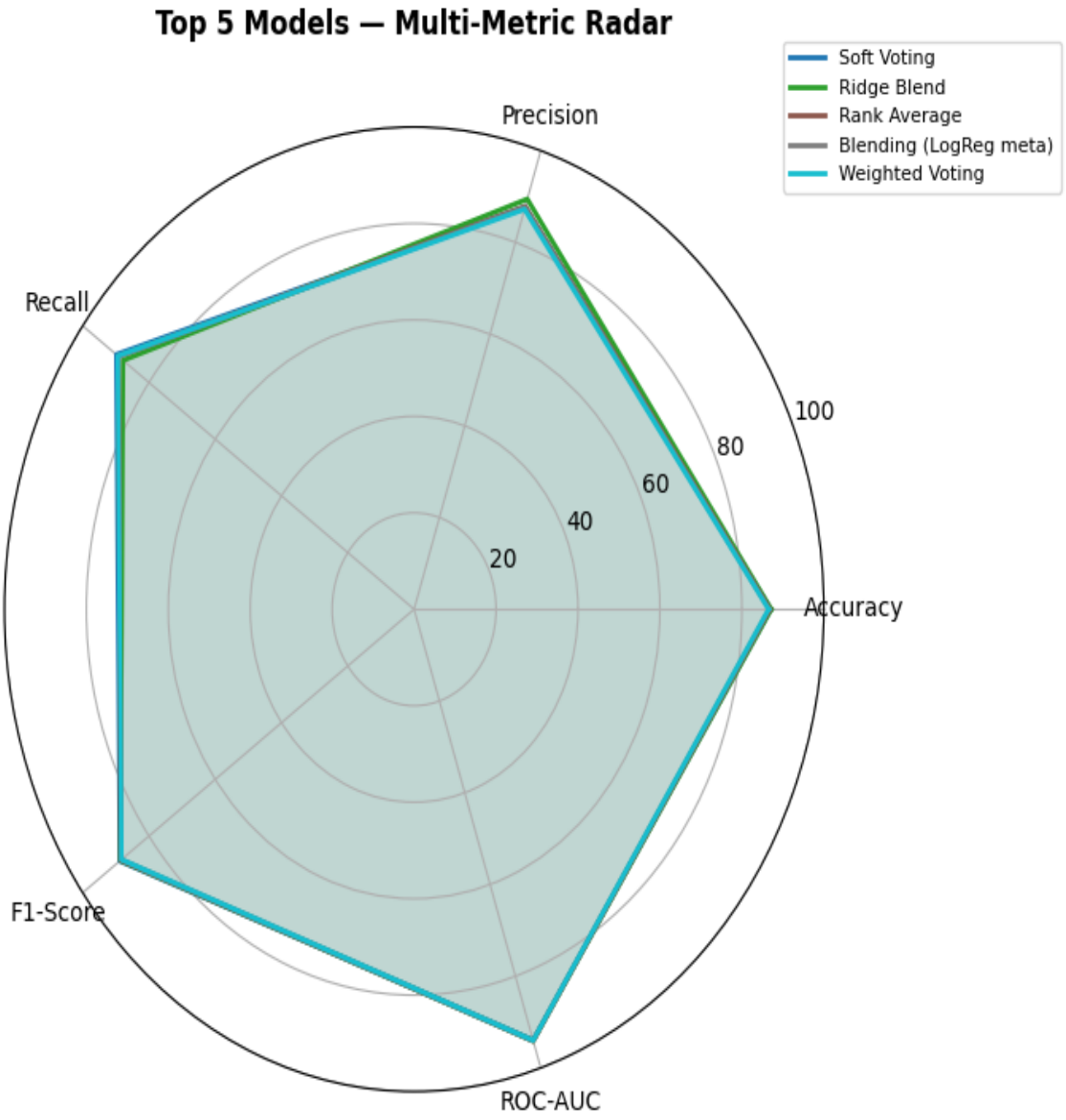


Figure 4.27: Multi-metric radar comparison of the top five models (Accuracy, Precision, Recall, F1-score, ROC-AUC).

Five metrics Accuracy, Precision, Recall, F1-score, and ROC-AUC are mapped onto five equally spaced radial axes, and each of the top five strategies is drawn as a closed polygon connecting its score on every axis. Because all five polygons trace nearly identical paths and are difficult to distinguish visually even at close inspection, the figure itself constitutes qualitative visual evidence for the statistical near-equivalence formally established by the McNemar's-test results in Figure 4.20.

The five polygons are visually almost indistinguishable across all five metrics, corroborating the McNemar's-test finding (Figure 4.20) that these strategies are statistically near-equivalent rather than meaningfully ranked.

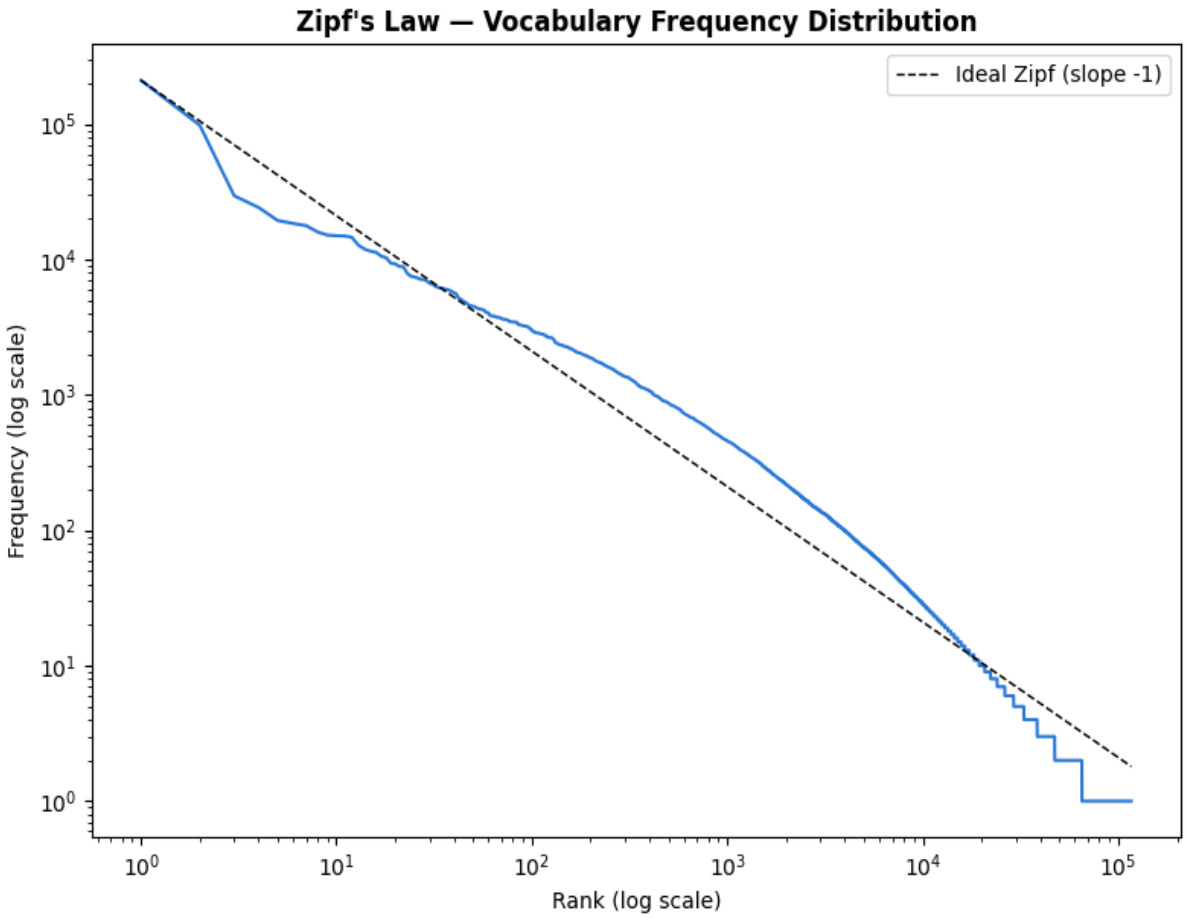


Figure 4.28: Zipf's-law check: word-rank versus frequency for the cleaned corpus vocabulary, log–log scale.

Word rank and raw frequency are both plotted on logarithmic axes, so that the inverse power-law relationship predicted by Zipf's law frequency proportional to the inverse of rank appears as a straight line of slope −1, shown here as a dashed reference. The cleaned corpus's empirical curve tracks this reference closely through the bulk of the vocabulary before departing from it at the extreme high- and low-frequency tails, a departure pattern typical of finite natural-language corpora and not indicative of any residual data-quality problem specific to this dataset.

The empirical curve tracks the ideal slope −1 closely through the middle frequency range, flattening at the highest ranks and steepening at the tail standard deviations for natural-language corpora, confirming the cleaned corpus behaves as genuine language text rather than degenerate or anomalous text.

4.13 Summary

This chapter has presented the complete experimental results for all twenty-seven models evaluated in this thesis. The central empirical finding is that a diversity-driven Soft-Voting ensemble, combining gradient-boosting, linear, recurrent-attention, and transformer base models, achieves the strongest overall performance (87.09% accuracy,

88.79% F1-score, 93.96% ROC-AUC), outperforming every individual model, including the strongest single classifier, LightGBM (86.40% F1). Feature-importance and explainability analysis (SHAP, LIME, and attention inspection) confirmed that this performance is driven by genuine linguistic and stylometric signal rather than by the collection artefacts identified and removed in Chapter 3, and error analysis indicated that a meaningful portion of the residual error reflects irreducible label noise rather than model shortcoming. Chapter 5 now examines these findings from the perspective of generalisation, overfitting behaviour, and the broader limitations and ethical considerations that must accompany any real-world application of this system.

CHAPTER 5
LIMITATIONS AND ETHICAL CONSIDERATIONS

5.1 Introduction

While Chapter 4 demonstrated that the proposed pipeline achieves a strong, leakage-aware performance ceiling of approximately 87–89% F1-score, it is essential to situate that result within an honest account of the study's boundaries. This chapter examines the limitations of the dataset, the modelling approach, and the evaluation protocol, before turning to the ethical considerations that must accompany any real-world deployment of an automated fake-news classifier. This discussion is presented deliberately as a distinct chapter, rather than as a brief closing caveat, in keeping with this thesis's broader emphasis (Section 1.5, Section 2.9) on methodological honesty over inflated claims of performance.

5.2 Limitations of the Study

5.2.1 Dataset Size and Domain Coverage

At 10,274 clean articles, the corpus used in this thesis is substantial for a self-constructed, individually audited Bengali dataset, but remains modest by comparison with the ~50,000-article BanFakeNews corpus [8] and far smaller than the hundred-thousand-plus-example datasets typically used to train English-language transformer-based classifiers from scratch. Section 4.4 and Section 4.5 showed directly that this constraint disadvantages both the deep-learning architectures (Section 3.7) and the transformer models (Section 3.8) relative to classical, feature-rich approaches. Domain coverage is similarly uneven: the pronounced category skew documented in Section 4.2.2 means that categories such as Technology, Crime, and Finance are represented by comparatively few training examples, and the category-wise error analysis of Section 4.9 shows this translates into measurably lower reliability for under-represented categories.

A related limitation is temporal coverage: the corpus is a snapshot, and both journalistic style and misinformation tactics evolve over time, so no guarantee of continued accuracy against future campaigns follows from this evaluation motivating the longitudinal-evaluation direction in Section 6.6.5.

5.2.2 Text-only Modality Limitation

Consistent with the scope defined in Section 1.6.1.1, the system processes only the textual content of news articles. Modern misinformation frequently relies on manipulated, mis-captioned, or out-of-context images and video accompanying an otherwise plausible textual narrative; such multimodal misinformation cannot be detected by a text-only pipeline of the kind developed here, regardless of how strong its textual performance is. This is a structural limitation rather than one that could be resolved through further tuning

of the current architecture, and it is the primary motivation for the multimodal future-work direction discussed in Section 6.6.2.

This is compounded when a manipulated image or video is paired with linguistically unremarkable captioning, since the deceptive content resides in the visual channel and the stylometric cues this thesis relies on (Section 3.5.3) are not designed to detect it.

5.2.3 Dialectal and Benglish Variation Challenges

As anticipated in Section 1.6.2.3 and Section 1.6.2.5, the corpus is dominated by formally written, standard Bengali news prose. Regional dialectal variation and, especially, "Benglish" Romanised or English–Bengali code-mixed text common on social media (Section 1.2.2) are essentially absent from the training data. Because the vocabulary underlying both the TF-IDF representations (Section 3.5.1–3.5.2) and the neural tokenisers (Section 3.7) was learned almost exclusively from standard Bengali script, the trained models should be expected to generalise poorly if applied directly to informal, code-mixed social-media text, even though such text is, in practice, a major vector for the spread of misinformation in Bangladesh (Section 1.1.2).

Because standard news prose and informal, code-mixed social-media text differ systematically in vocabulary and orthography, this gap is unlikely to close through simply collecting more of the same kind of data it would require a separate, targeted code-mixed data-collection effort.

5.2.4 Computational Constraints of Transformer Models

Fine-tuning BanglaBERT and XLM-RoBERTa (Section 3.8) required GPU acceleration, mixed-precision training, gradient accumulation, and dynamic padding simply to complete training within a practical time budget (Section 3.10.4); inference at scale would similarly require GPU resources that are not universally available to the smaller newsrooms, fact-checking organisations, and civil-society groups (Section 1.5.3, Section 1.5.5) this research is ultimately intended to support. This computational asymmetry is one of the reasons this thesis explicitly highlights the gradient-boosting models LightGBM, XGBoost, and CatBoost as the most realistic candidates for lightweight, resource-constrained deployment (Section 1.2.4, Section 5.8.5 in the concluding chapter), notwithstanding the fact that the very best result overall requires the heavier ensemble combining transformer and classical predictions.

This asymmetry matters for equitable access: per Table 4.1 and Figure 4.26, the lightweight, CPU-trainable gradient-boosting models alone stay within roughly two to three F1 points of the full ensemble at a fraction of the computational cost a more realistic deployment path for smaller newsrooms and fact-checkers (Section 1.5.3, 1.5.5).

5.3 Ethical Considerations

5.3.1 Data Privacy and Source Attribution

The dataset used in this research consists of published news content rather than private user data, which substantially reduces direct privacy risk relative to, for example, a system trained on private social-media messages. Nonetheless, responsible use of this dataset and any derived model requires clear attribution of source material, avoidance of redistribution that could infringe the original publishers' rights, and careful handling of the metadata (source, category, date) attached to each article, particularly should this pipeline later be extended toward the source-level or outlet-level analysis that Section 3.3.4's leakage audit deliberately avoided building into the classifier itself.

5.3.2 Bias and Fairness in Predictions

Any model trained on historical, human-curated data risks inheriting the biases present in that data. In this study, the most concrete manifestation of this risk is the category imbalance documented in Section 4.2.2 and its consequence for category-wise reliability (Section 4.9): a deployed system would currently be expected to perform less reliably, and therefore to be less trustworthy, when applied to Technology, Crime, or Finance news than to National or International news, simply because of how the training corpus happens to be distributed. A second, more subtle fairness risk is that models could, in principle, learn to associate a particular publication style, region, or topic with the Fake label for reasons unrelated to actual veracity; the explicit shortcut-detection and removal process of Section 3.3.4–3.4.3, and its empirical validation through the feature-importance analysis of Section 4.8, is this thesis's primary safeguard against exactly this failure mode, though it cannot guarantee the complete absence of subtler, unmeasured biases.

5.3.3 Responsible Use of Predictions

The models developed in this thesis output a probability estimate, not a certified verdict, and as the error analysis of Section 4.9 makes concrete, with roughly one in ten to one in six articles misclassified depending on category and length they are demonstrably fallible. Consistent with the intended use cases described in Section 1.5 (decision support for fact-checkers, pre-filtering for newsroom moderation teams, triage for platform-level review), predictions from this system should be treated strictly as a supportive signal to inform human review, never as an automated basis for removing content, penalising a publisher, or labelling an author, particularly given the evidence from Section 4.9 that a meaningful share of “errors” may in fact reflect disagreement with noisy or contestable ground-truth labels rather than a genuine model failure.

5.3.4 Transparency and Explainability

In direct support of responsible use, this thesis deliberately invested in model-agnostic and model-intrinsic explainability analysis LightGBM and Logistic-Regression feature importance, global SHAP analysis, local LIME explanations, and attention-weight inspection (Section 4.8) rather than treating the best-performing ensemble as an

unexamined black box. Any future deployment of this system should preserve and surface this explainability layer to end users (fact-checkers, moderators, or readers), so that a given prediction can always be accompanied by a human-interpretable account of which words or features drove it, consistent with growing regulatory and professional expectations around explainable AI in content-moderation contexts.

5.4 Summary

This chapter has situated the results of Chapter 4 within an honest account of this study's limitations dataset size and domain coverage, text-only modality, dialectal and code-mixed language coverage, and the computational demands of transformer-based components and of the ethical considerations that must govern any real-world use of the resulting system: privacy and attribution, bias and fairness, responsible deployment as a decision-support rather than an autonomous-judgement tool, and sustained transparency through explainability. These considerations are not presented as an afterthought but as integral to a defensible claim of scientific and practical validity. Chapter 6 concludes the thesis, synthesising the key findings from Chapters 4 and 5 into a final statement of contribution and a concrete agenda for future research.

CHAPTER 6
CONCLUSION AND FUTURE WORK

6.1 Introduction

This final chapter draws together the findings of Chapters 3–5 into a concise statement of what this thesis set out to do, what it found, what it contributes, and where the most promising directions for future research lie. Section 6.2 summarises the research work as a whole; Section 6.3 distils the key findings; Section 6.4 and Section 6.5 restate, respectively, the practical implications and the specific contributions of the study; Section 6.6 sets out six concrete future-work directions; and Section 6.7 closes the thesis with final concluding remarks.

6.2 Summary of Research Work

This research addressed the problem of automatically detecting fake news in Bengali, a low-resource language for which existing detection tools and research remain comparatively scarce despite the severe, well-documented real-world consequences of unchecked misinformation in Bengali-speaking communities (Chapter 1). Building on a review of both the global fake-news-detection literature and the small existing body of Bengali-specific work (Chapter 2), this thesis constructed and rigorously audited an 11,839-article Bengali news corpus, reducing it to 10,274 clean articles through systematic deduplication, label-conflict resolution, and the removal of collection-artefact shortcuts that had been found to correlate with the class label in up to 99.6% of affected instances (Chapter 3). Twenty-seven models spanning classical machine learning, deep learning, transformer fine-tuning, and eight ensemble-combination strategies were then trained and compared under an identical, strictly leakage-free 70/15/15 protocol (Chapter 3), and the resulting performance was analysed in depth, including exploratory data analysis, comparative benchmarking, feature-importance and explainability analysis (SHAP, LIME, attention), and category- and length-stratified error analysis (Chapter 4). The study concluded with a critical examination of its limitations and the ethical considerations attaching to real-world deployment (Chapter 5).

6.3 Key Findings

6.3.1 Model Performance Insights

A diversity-driven Soft-Voting ensemble, combining Gradient Boosting, CatBoost, XGBoost, Extra Trees, Linear SVM, BiLSTM-Attention, and XLM-RoBERTa as base learners, achieved the strongest performance of all twenty-seven models evaluated: 87.09% accuracy, 87.75% precision, 89.85% recall, 88.79% F1-score, and 93.96% ROC-AUC on the held-out test set. Gradient-boosting classical models (LightGBM, CatBoost, XGBoost) were the strongest individual model family, consistently outperforming every deep-learning architecture and both fine-tuned transformers, an outcome attributable to

the modest scale of the available training data (7,191 articles) relative to what data-hungry deep and transformer architectures typically require to reach their full potential.

6.3.2 Feature Contribution Insights

Convergent evidence from LightGBM gain-based importance, Logistic-Regression coefficients, and global SHAP analysis identified character entropy, lexical diversity, headline–content overlap, and journalistic-rigour markers (quotation-attribution verbs, named designations, dates, and organisation references) as the most consistently influential predictors genuine stylometric and content-based cues rather than collection artefacts, a conclusion reinforced by the deliberate absence of any previously identified shortcut token among the most important features.

6.3.3 Dataset-related Insights

The data-quality audit conducted in this thesis (Section 3.3) uncovered issues 1,247 exact duplicate rows, 310 rows carrying conflicting labels, and outlet/scraping-artefact shortcuts correlating with the label at up to 99.6% that are not reported as having been addressed in the two Bengali fake-news studies analysed in the greatest depth for this thesis (Table 2.1). This indicates that a substantial share of the very high accuracy figures reported elsewhere in the literature may be attributable, at least in part, to leakage effects of exactly this kind, and that the ~87–89% F1 ceiling established in this thesis, while numerically lower, is the more scientifically defensible estimate of genuine Bengali fake-news-detection capability under the conditions studied here.

6.4 Practical Implications

The findings of this research carry direct practical relevance for the stakeholders identified in Section 1.5. For fact-checking organisations (Section 1.5.5), the trained ensemble can serve as a triage layer, directing scarce human verification capacity toward the articles the model flags as most likely fabricated. For newsrooms and digital publishers (Section 1.5.3), the lightweight gradient-boosting models identified in Section 4.3 offer a practical, computationally modest starting point for in-house content-verification tooling, without requiring the GPU infrastructure that the transformer and full-ensemble configurations demand. For platform-level content-moderation teams and policy researchers (Section 1.5.2), the explicit demonstration that leakage-aware evaluation materially changes reported performance (Section 4.10, Section 6.3.3) is itself a practically important methodological lesson: any procurement or adoption decision regarding a third-party Bengali fake-news detector should require evidence of an equivalent leakage audit before its reported accuracy figures are taken at face value.

6.5 Contributions of the Study

This thesis makes five specific contributions to the field of Bengali fake-news detection and, more broadly, to leakage-aware evaluation practice in low-resource NLP:

11. A rigorous, transparently documented data-quality audit methodology covering duplicate detection, label-conflict resolution, and collection-artefact shortcut analysis applied to a Bengali fake-news corpus, together with concrete evidence (a 99.6% shortcut correlation) of how severely such issues can distort reported performance if left unaddressed.
12. An independently constructed, deduplicated, near-balanced 10,274-article Bengali news corpus spanning twelve categories, suitable for future leakage-aware benchmarking.
13. The most comprehensive leakage-free comparison identified in the current Bengali fake-news literature, spanning twenty-seven models across four methodological paradigms (classical machine learning, deep learning, transformer fine-tuning, and ensemble learning) under one identical experimental protocol.
14. A diversity-driven ensemble methodology, with automatic, correlation-aware base-model selection, that achieves the best documented honest performance on this dataset (88.79% F1) and provides a template applicable to other low-resource text-classification tasks.
15. A multi-method explainability analysis (feature importance, SHAP, LIME, and attention inspection) that verifies, rather than merely assumes, that the resulting classifier relies on genuine linguistic and journalistic-rigour signal rather than on artefacts of the data-collection process.

6.6 Future Work

Building directly on the limitations identified in Chapter 5, six concrete directions for future research are proposed.

6.6.1 Larger and More Diverse Dataset Collection

The single most promising direction for improving genuine detection performance, as opposed to merely reported performance, is expanding the training corpus, whether through additional independently collected and audited data or through transfer learning from a larger existing resource such as BanFakeNews [8], while carrying forward this thesis's leakage-prevention protocol (Section 3.3) rather than adopting a larger corpus's data uncritically.

6.6.2 Multimodal Detection (Text, Image, Video)

Extending the pipeline to incorporate accompanying images and video through, for example, reverse-image-search-style manipulation detection or joint text–image embedding models would directly address the text-only limitation identified in Section 5.2.2 and better reflect the multimodal nature of real-world Bengali misinformation described in Section 1.1.3.

6.6.3 Real-time Prediction Systems

Converting the offline batch pipeline validated in this thesis into a streaming, low-latency service building on the lightweight gradient-boosting models identified as the most deployment-practical individual classifiers (Section 4.3, Section 5.2.4) would directly address the real-time detection gap discussed in Section 1.2.4 and move this research closer to the fact-checking and platform-moderation use cases envisioned in Section 1.5.

6.6.4 Bias Mitigation and Ethical AI

Future work should extend the fairness analysis of Section 5.3.2 with formal bias-auditing techniques across news category, publication source, and topic, and explore targeted data augmentation or re-weighting strategies to reduce the category-wise reliability gap documented in Section 4.9, while preserving the explainability infrastructure (Section 4.8, Section 5.3.4) already built into this pipeline.

6.6.5 Longitudinal Performance Analysis

Because fake-news tactics evolve over time (Section 1.2.3), a valuable extension would track model performance longitudinally against newly published Bengali news, testing whether the linguistic cues identified as important in Section 4.8 (and, correspondingly, this thesis's core methodological finding the value of leakage-aware evaluation) remain stable as misinformation strategies change, or whether periodic retraining and re-auditing become necessary.

6.6.6 Deployment in Education Policy and as a Service

Beyond the research pipeline itself, translating the best-performing configuration into a deployable browser extension or API service potentially exposing the explainability layer (Section 4.8) directly to end users and engaging with fact-checking organisations and media-literacy initiatives (Section 1.5.5) on its evaluation in real operational settings would test whether the leakage-aware performance ceiling established in this thesis (~87–89% F1) translates into genuine, measurable benefit outside a controlled research environment.

6.7 Concluding Remarks

Table 6.1: Mapping of research questions (Chapter 1) to key findings (Chapters 4–5)

Research Question (Ch. 1)	Key Finding (Ch. 4–5)
RQ1 — Effective preprocessing	NFC normalisation, artefact/outlet-marker removal, and avoidance of unreliable stemming jointly prevent shortcut learning while preserving genuine stylometric signal (Sections 4.8, 4.10, 4.11.1).
RQ2 — Optimal feature extraction	No single representation suffices; the strongest results combine word/character TF-IDF, hand-crafted meta-features, and transformer embeddings at the ensemble level (Section 4.11.2).
RQ3 — Deep learning vs classical ML	Classical, feature-rich models (especially gradient boosting) outperform deep architectures trained from scratch at this corpus scale, consistent with overfitting evidence in Figure 4.7 (Section 4.11.3).
RQ4 — Effectiveness of BanglaBERT	BanglaBERT under-performs XLM-RoBERTa and all classical/ensemble models, indicating that fine-tuning-data scale currently outweighs the benefit of Bengali-specific pretraining (Section 4.11.4).
RQ5 — Genuine linguistic cues	Entropy, lexical diversity, headline–content overlap, and journalistic-rigour markers are the consistent top predictors across three independent explainability methods (Section 4.11.5).

Table 6.1 closes the results-and-discussion arc opened in Section 1.4 by pairing each research question with the specific Chapter 4/Chapter 5 evidence that answers it, so that a reader who consulted only Chapter 1 and this closing table, without the intervening chapters, could still reconstruct the thesis's core empirical claims and locate the section in which each is substantiated in full.

This thesis set out to determine whether an effective, trustworthy automated system for detecting Bengali fake news could be built while directly addressing a methodological weakness identified in the literature: data leakage through duplication and collection artefacts (Section 2.9). After de-duplication, label-conflict resolution, and removal of collection-artefact shortcuts, twenty-seven models spanning classical machine learning, deep learning, transformer fine-tuning, and ensemble learning were compared under an identical, leakage-free protocol. A diversity-driven Soft-Voting ensemble achieved the best result, with 87.09% accuracy and 88.79% F1-score, outperforming every individual model. Because this result avoids the leakage shortcuts shown to inflate accuracy on comparable scraped corpora, it provides a more trustworthy estimate of real-world Bengali fake-news-detection performance. The analysis further establishes that dataset size and residual label noise, rather than model architecture, are the primary factors limiting performance beyond the ~87–89% F1 ceiling. Future research should therefore priorities dataset expansion, multimodal detection, and real-world deployment testing. More broadly, this thesis demonstrates that for Bengali and other low-resource languages facing similar data-collection challenges, a rigorous, transparently reported, leakage-aware evaluation protocol is a precondition for building fake-news-detection systems that can be trusted beyond the research environment.

REFERENCES

References are numbered in the order in which they are first cited in the text, following IEEE style.

[1] Bangladesh Telecommunication Regulatory Commission (BTRC) and Bangladesh Bureau of Statistics (BBS), "Internet and Mobile Phone Subscribers in Bangladesh," Statistical Reports, 2025.

[2] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake News Detection on Social Media: A Data Mining Perspective," ACM SIGKDD Explorations Newsletter, vol. 19, no. 1, pp. 22–36, 2017.

[3] X. Zhou and R. Zafarani, "A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities," ACM Computing Surveys, vol. 53, no. 5, pp. 1–40, 2020.

[4] Reuters, BBC News, and Human Rights Watch, news reporting and investigative coverage of the 2012 Ramu and October 2021 Cumilla communal-violence incidents in Bangladesh, 2012–2021.

[5] Human Rights Watch and Reuters, news reporting and investigative coverage of the 2019 child-abduction rumor-driven mob killings in Bangladesh (Padma Bridge rumors), 2019.

[6] W. Y. Wang, "'Liar, Liar Pants on Fire': A New Benchmark Dataset for Fake News Detection," in Proc. 55th Annual Meeting of the Association for Computational Linguistics (ACL), 2017, pp. 422–426.

[7] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, "FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News on Social Media," Big Data, vol. 8, no. 3, pp. 171–188, 2020.

[8] M. Z. Hossain, M. A. Rahman, M. S. Islam, and S. Kar, "BanFakeNews: A Dataset for Detecting Fake News in Bangla," in Proc. 12th Language Resources and Evaluation Conference (LREC), 2020, pp. 2862–2871.

[9] "Enhancing Bangla Fake News Detection Using Bidirectional Gated Recurrent Units and Deep Learning Techniques," ACM, 2024. doi: 10.1145/3659677.3659703. [Author names as per publication record; to be verified against the source PDF prior to final submission.]

[10] "A Sophisticated Feature Vectorization-based Stacked Machine Learning Approach for Fake News Detection in Bangla and English," Social Network Analysis and Mining, Springer, vol. 15, 2025. doi: 10.1007/s13278-025-01552-6. [Author names as per publication record; to be verified against the source PDF prior to final submission.]

[11] A. Bhattacharjee, T. Hasan, W. U. Ahmad, K. S. Mubasshir, M. S. Islam, A. Iqbal, M. S. Rahman, and R. Shahriyar, "BanglaBERT: Language Model Pretraining and Benchmarks

for Low-Resource Language Understanding Evaluation in Bangla," in Findings of the Association for Computational Linguistics: NAACL 2022, pp. 1318–1327.

[12] S. Sarker, "BanglaBERT: Bengali Mask Language Model for Bengali Language Understanding," HuggingFace Model Card (sagorsarker/bangla-bert-base), 2020.

[13] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised Cross-lingual Representation Learning at Scale," in Proc. 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020, pp. 8440–8451.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 5998–6008.

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," in Proc. ICLR Workshop Track, 2013.

[17] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," Transactions of the Association for Computational Linguistics (TACL), vol. 5, pp. 135–146, 2017.

[18] G. Salton and C. Buckley, "Term-weighting Approaches in Automatic Text Retrieval," Information Processing & Management, vol. 24, no. 5, pp. 513–523, 1988.

[19] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[20] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119–139, 1997.

[21] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794.

[22] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 3146–3154.

[23] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased Boosting with Categorical Features," in Advances in Neural Information Processing Systems (NeurIPS), 2018, pp. 6638–6648.

[24] D. H. Wolpert, "Stacked Generalization," Neural Networks, vol. 5, no. 2, pp. 241–259, 1992.

[25] R. Caruana, A. Niculescu-Mizil, G. Crew, and A. Ksikes, “Ensemble Selection from Libraries of Models,” in Proc. 21st International Conference on Machine Learning (ICML), 2004.

[26] C. Cortes and V. Vapnik, “Support-Vector Networks,” Machine Learning, vol. 20, no. 3, pp. 273–297, 1995.

[27] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning Phrase Representations Using RNN Encoder–Decoder for Statistical Machine Translation,” in Proc. EMNLP, 2014, pp. 1724–1734.

[29] Y. Kim, “Convolutional Neural Networks for Sentence Classification,” in Proc. EMNLP, 2014, pp. 1746–1751.

[30] D. Bahdanau, K. Cho, and Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate,” in Proc. ICLR, 2015.

[31] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 4765–4774.

[32] M. T. Ribeiro, S. Singh, and C. Guestrin, “‘Why Should I Trust You?’: Explaining the Predictions of Any Classifier,” in Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135–1144.

[33] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine Learning in Python,” Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[34] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” in Advances in Neural Information Processing Systems (NeurIPS), 2019, pp. 8026–8037.

[35] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al., “Transformers: State-of-the-Art Natural Language Processing,” in Proc. EMNLP: System Demonstrations, 2020, pp. 38–45.

APPENDIX

Appendix A Sample Dataset Records

Table A.1 reproduces five representative rows drawn directly from `pial_bengali_fake_news_dataset.xlsx`, illustrating the four raw fields (`category`, `headline`, `content`, `label`) described in Section 3.3.1. Content fields are truncated for presentation; full records are retained in the source dataset file. The fourth row illustrates a Fake-labelled article carrying the outlet marker "Bengal Beats" discussed in Section 3.3.4 and Section 3.4.3, shown here in its raw, pre-cleaning form for illustrative purposes.

Table A.1: Representative raw dataset records (category, headline, truncated content, label)

Category	Headline	Content (truncated)	Label
Education	১৪তম শিক্ষক নিবন্ধনের চূড়ান্ত ফল এ মাসে	চলতি মাসে ১৪তম শিক্ষক নিবন্ধন পরীক্ষার চূড়ান্ত ফলাফল প্রকাশিত হবে… (1,180 characters, truncated)	Real (1)
National	মুন্সীগঞ্জে প্লাস্টিক কারখানায় আগুন	মুন্সীগঞ্জ সদরের কমলাঘাট বাণিজ্যিক বন্দরের একটি প্লাস্টিক কারখানাসহ… (620 characters, truncated)	Real (1)
National	নাগেশ্বরীতে ফেনসিডিলসহ দম্পতি আটক	কুড়িগ্রাম: কুড়িগ্রামের নাগেশ্বরী উপজেলায় ৩৭ বোতল ফেনসিডিলসহ এক দম্পতিকে… (590 characters, truncated)	Real (1)
Miscellaneous	লোহার খনিতে কি থাকে জিজ্ঞেস করে থাপ্পড় খেলেন এক সাংবাদিক - Bengal Beats	সম্প্রতি বাংলাদেশে একটি লোহার খনি আবিষ্কার হয়েছে… (satirical / fabricated narrative; outlet marker 'Bengal Beats' present in raw headline before cleaning — Section 3.4.3)	Fake (0)
National	কামাল-বি. চৌধুরী নিয়ে মাথাব্যথা নেই	ড. কামাল হোসেন ও বদরুদ্দোজা চৌধুরীর নেতৃত্বে নতুন ঐক্য গড়ার যে প্রক্রিয়া চলছে… (2,050 characters, truncated)	Fake (0)

The five rows reproduced in Table A.1 are chosen to illustrate range rather than to be representative in a statistical sense: they span three categories (Education, National, Miscellaneous), both class labels, and, in the fourth row, a Fake-labelled article still carrying its original outlet marker, included deliberately in raw, pre-cleaning form so that the shortcut pattern described narratively in Section 3.3.4 can be inspected directly rather than taken only on the strength of the aggregate 99.6% correlation figure reported there.

Appendix B Representative Code Excerpts

The following excerpts, reproduced from the experimental notebook underlying this thesis, illustrate three representative implementation details discussed narratively in Chapter 3.

B.1 Outlet-marker and Scraping-artifact Shortcut Detection (Section 3.3.4)

```
OUTLET_MARKERS = r"মতিকণ্ঠ|জাগো|Bengal|Beats|দৈনিক"
SCRAPE_TEST    = r"ShareTweet|adsbygoogle|advertisement"
hm = df["headline"].str.contains(OUTLET_MARKERS, regex=True)
```

```
hj = df["content"].str.contains(SCRAPE_TEST, regex=True, case=False)
print(f"Outlet-marker in headline: {hm.sum():4d} rows | fake-rate
{df.loc[hm,'label'].eq(0).mean()*100:.1f}%")
print(f"Scrape-junk in content   : {hj.sum():4d} rows | fake-rate
{df.loc[hj,'label'].eq(0).mean()*100:.1f}%")
# => stripped as noise in text cleaning (NOT used as features).
```

B.2 Word- and Character-level TF-IDF Construction (Section 3.5.1–3.5.2, 3.5.6)

```
tfidf_word = TfidfVectorizer(max_features=30000, ngram_range=(1,3),
                             sublinear_tf=True, min_df=3, max_df=0.9)
tfidf_char = TfidfVectorizer(max_features=20000, ngram_range=(2,5),
sublinear_tf=True,
                             min_df=3, max_df=0.9, analyzer="char_wb")

Xw_tr = tfidf_word.fit_transform(text_full.loc[idx_tr])
Xw_va = tfidf_word.transform(text_full.loc[idx_va])
Xw_te = tfidf_word.transform(text_full.loc[idx_te])
# ... (character-level matrix built identically) ...

scaler = StandardScaler()
mt = csr_matrix(scaler.fit_transform(meta_df.loc[idx_tr].values))   # fit on
TRAIN only
mv = csr_matrix(scaler.transform(meta_df.loc[idx_va].values))
me = csr_matrix(scaler.transform(meta_df.loc[idx_te].values))

X_tr = hstack([Xw_tr, Xc_tr, mt]).tocsr()   # ~50,000-dimension combined matrix
```

B.3 Diversity-Aware Automatic Base-Model Selection (Section 3.9)

```
mf1 = {m: f1_score(y_val, (val_probs[m] >= 0.5).astype(int)) for m in
val_probs}
ranked = sorted(mf1, key=mf1.get, reverse=True)
selected = [ranked[0]]
for m in ranked[1:]:
    if max(np.corrcoef(vdf[m], vdf[s])[0, 1] for s in selected) < 0.97 and
mf1[m] > 0.55:
        selected.append(m)
    if len(selected) >= 7:
        break
```

B.4 Validation-only Weighted-Voting Optimisation (Section 3.9.3)

```
def opt_weights(names, n=3000, seed=42):
    P = np.column_stack([val_probs[x] for x in names])
    rng = np.random.default_rng(seed); bw, bf = None, 0
    for _ in range(n):
```

```
        w = rng.dirichlet(np.ones(len(names)))
        f = f1_score(y_val, ((P @ w) >= 0.5).astype(int))
        if f > bf:
            bf, bw = f, w
    return dict(zip(names, bw)), bf

weights, _ = opt_weights(selected)   # tuned on VALIDATION set only
```

B.5 Bengali-Font-Aware Figure Rendering (referenced by Figures 4.9 and 4.12)

Figures 4.9 and 4.12 were generated before a Bengali-capable font with correct glyph shaping (conjunct consonants, vowel-sign reordering) was integrated into the plotting environment. The corrected procedure below downloads Noto Sans Bengali, verifies Raqm complex-text-shaping support via Pillow, and should be used to regenerate fully-labelled versions of both figures before submission.

```
import os, urllib.request
from PIL import Image, ImageDraw, ImageFont, features

font_path = "NotoSansBengali-Regular.ttf"
if not os.path.exists(font_path):
   urllib.request.urlretrieve(
      "https://github.com/googlefonts/noto-fonts/raw/main/hinted/ttf/"
      "NotoSansBengali/NotoSansBengali-Regular.ttf", font_path)

has_raqm = features.check("raqm")
layout = ImageFont.Layout.RAQM if has_raqm else ImageFont.Layout.BASIC
font_label = ImageFont.truetype(font_path, 32, layout_engine=layout)

word_char_names = set(tfidf_word.get_feature_names_out()) |
set(tfidf_char.get_feature_names_out())
imp_ngrams_only = imp[imp.index.isin(word_char_names)]
top_imp = imp_ngrams_only.head(25).sort_values(ascending=True)

img = Image.new("RGB", (1600, 1200), "white")
draw = ImageDraw.Draw(img)
for i, (feature, value) in enumerate(top_imp.items()):
   y = 150 + i * 38
   draw.text((20, y), str(feature), font=font_label, fill="black")
   bar_len = int((value / top_imp.max()) * 600)
   draw.rectangle([(550, y + 8), (550 + bar_len, y + 35)], fill=(42, 120, 214))
img.save("fig_feature_importance_final.png")
```

B.6 Handcrafted Meta-Feature Extraction (Section 3.5.3, Figure 4.15)

The following excerpt implements a representative subset of the meta-features visualised by class in Figure 4.15 — lexical-diversity/entropy, headline–content overlap, and journalistic-rigour feature groups.

```
QUOTE = r"বলেন|জানান|জানিয়েছেন|মন্তব্য|উল্লেখ করেন|দাবি করেন|স্বীকার করেন"
DESIG = r"কর্মকর্তা|চেয়ারম্যান|মুখপাত্র|সচিব|পরিদর্শক|মন্ত্রী|চিকিৎসক|বিশেষজ্ঞ"

def extract_meta(d):
    m = pd.DataFrame(index=d.index)
    words = d["content_clean"].str.split()
    m["ct_words"]     = words.apply(len)
    m["unique_ratio"] = words.apply(lambda w: len(set(w)) / max(len(w), 1))
    m["entropy"]      = d["content_clean"].apply(shannon_entropy)
    m["hl_ct_overlap"] = [jaccard(h, c) for h, c in zip(d["headline_clean"],
d["content_clean"])]
    m["quote_count"]  = d["content_clean"].str.count(QUOTE)
    m["desig_count"]  = d["content_clean"].str.count(DESIG)
    m["digit_density"] = d["content_clean"].apply(lambda t: sum(c.isdigit() for c in t) /
max(len(t), 1))
    return m
```

Appendix C Supplementary Hyperparameter Summary

Table C.1 consolidates the key hyperparameters referenced throughout Chapter 3 into a single reference table.

Table C.1: Consolidated hyperparameter and configuration summary

Component	Configuration
Word TF-IDF	ngram_range=(1,3), max_features=30000, sublinear_tf=True, min_df=3, max_df=0.9
Character TF-IDF	analyzer='char_wb', ngram_range=(2,5), max_features=20000, sublinear_tf=True
Chi2 feature selection (tree models)	SelectKBest(chi2, k=8000)
RBF-SVM / KNN dimensionality reduction	TruncatedSVD, n_components=300
Linear SVM	CalibratedLinearSVC, max_iter=20000
Deep-learning embeddings	EMBED_DIM=120, dropout=0.55, MAX_LEN=400, VOCAB_SIZE=40000
Deep-learning training	class-weighted loss, gradient clipping, FP16 mixed precision, ReduceLROnPlateau, early stopping on val-F1, weight_decay=3e-4
Train-only augmentation	35% of training rows; random word deletion (p=0.08) / sentence-order shuffling
Transformer fine-tuning	backbone LR=2e-5, head LR=1e-4, 6% warm-up, weight_decay=0.01, gradient accumulation (effective batch = 32), dynamic padding, 12 epochs
Weighted-voting search	Dirichlet random search, n=3000 trials, optimised on validation F1

Base-model diversity filter	pairwise prediction correlation < 0.97; max 7 base models
K-Fold stacking	5-fold out-of-fold predictions (LR, SVM, NB, LightGBM); LR meta-model

Consolidating these configuration values in one place, rather than leaving them distributed across Sections 3.5–3.9 alone, is intended to support exact reproducibility: each row of Table C.1 gives the literal parameter values needed to reconstruct the corresponding component of the pipeline without cross-referencing the narrative methodology text, complementing rather than duplicating the reasoning for each choice already given in Chapter 3.