DETECTING PROPAGANDA IN MARATHI NEWS ARTICLES USING MACHINE LEARNING AND DEEP LEARNING TECHNIQUES

A Thesis Submitted in partial Fullfilment of the
Requirements for the Dgree of
Master of Technology
in
Artificial Intelligence & Data Science Engineering

Submitted by
Student Name:- Manjul Mayank
Roll No:-(2303res27)
under the guidance of
Dr. Sriparna Saha

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY PATNA
PATNA - 800013,BIHAR
JUNE 2025

ACKNOWLEDGMENTS

I sincerely express my deepest gratitude to **Dr. Sriparna Saha**, my supervisor, for her invaluable guidance, constant support and insightful feedback throughout this research journey. I am forever grateful to my **mother**, whose unwavering love, sacrifices, and encouragement have been my greatest strength. Her endless patience and belief in me kept me motivated even during the most challenging times. My heartfelt thanks to my **brother** for his constant encouragement, understanding, and unwavering faith in my abilities, especially when I doubted myself. This achievement would not have been possible without their unconditional love and support.

Certificate

This is to certify that the thesis entitled **"Detecting Propaganda in Marathi News Articles Using Machine Learning and Deep Learning Techniques"**, submitted by **Manjul Mayank** to Indian Institute of Technology Patna, is a record of bonafide research work under my (our) supervision.

I (we) consider it worthy of consideration for the degree of Master of Technology of this Institute. This work or a part has not been submitted to any university/institution for the award of degree/diploma. The thesis is free from plagiarized material.

Sriparna Saha

Supervisor

22/06/2025

Date:

Declaration

I certify that

a. The work contained in this thesis is original and has been done by myself under the general supervision of my supervisor/s.

b. The work has not been submitted to any other Institute for degree or diploma.

c. I have followed the Institute norms and guidelines and abide by the regulation as given in the Ethical Code of Conduct of the Institute.

d. Whenever I have used materials (data, theory and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the reference section.

e. The thesis document has been thoroughly checked to exclude plagiarism.

Signature of the Student

Roll No: 2303res27

TABLE OF CONTENTS

Section **Page**

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

MRS	Marathi Readability Score
BiLSTM	Bidirectional Long Short-Term Memory
NLP	Natural Language Processing
TF-IDF	Term Frequency-Inverse Document Frequency
SVM	Support Vector Machine
MLP	Multilayer Perceptron
AUC	Area Under the Curve
MPropCorpus	Marathi Propaganda Corpus (dataset of 4,309 articles)
MarathiPropLex	Marathi Propaganda Lexicon (1,200 annotated terms)

ABSTRACT

The rapid proliferation of digital news consumption, particularly through social media platforms, has contributed significantly to the spread of misinformation and propaganda. While automated detection systems for English-language misinformation have advanced considerably, low-resource languages like Marathi remain underserved, despite having a large and influential speaker base. This thesis addresses the critical need for propaganda detection in Marathi by developing a linguistically-aware, machine learning-powered system capable of differentiating factual content from manipulative or propagandist narratives.

A custom web scraping framework was built to extract more than 4,000 Marathi news articles from reliable and suspicious sources such as Lokmat, NDTV Marathi, and FactCrescendo. A specialized Marathi text preprocessing pipeline was developed to handle Devanagari script normalization, tokenization, and context-aware stop word removal, especially targeting emotionally charged or propagandist phrases.

The study explores both traditional machine learning models (Logistic Regression, SVM, Random Forest, XGBoost, Naive Bayes, MLP) and a deep learning model using Bidirectional Long-Short-Term Memory (BiLSTM). All models were trained in TF-IDF and stylistic characteristics, while BiLSTM was trained on token sequences. The best performing models, XGBoost and BiLSTM, achieved near perfect results with accuracy, precision, recall, and F1 scores that exceeded 99. 5%.

In addition, a novel Marathi Readability Score (MRS) readability metric was introduced to capture the linguistic complexity and highlight the stylometric differences between propaganda and factual texts. The system was evaluated on real-world examples and demonstrated strong practical utility in classifying subtle and explicit propaganda with high confidence.

This work presents one of the first comprehensive efforts in computational propaganda detection for the Marathi language, contributing both technical innovations and publicly available resources for further research. It establishes a strong foundation for scalable misinformation detection systems in other low-resource Indian languages and offers a pathway for ethically grounded, culturally aware AI applications in journalism and public discourse.

Chapter 1

INTRODUCTION

1.1 Background and Motivation

The digital revolution has fundamentally transformed the way information is consumed and disseminated. With the rise of social media platforms and online news portals, millions now rely on digital media as their primary source of news. However, this democratization of information dissemination has led to a parallel increase in the spread of misinformation, fake news, and propaganda especially in regional languages.

While there has been significant research in propaganda detection for English and other high-resource languages, regional Indian languages such as Marathi remain critically underrepresented in this space. Given that Marathi is spoken by over 83 million individuals, primarily in the state of Maharashtra, the absence of robust detection tools poses a serious societal and technological gap.

1.1.1 The Growing Threat of Propaganda in Indian Languages

The digital footprint of the Marathi language has grown rapidly in recent years. According to the Indian Readership Survey (2022), there are over 15 million Marathi internet users, with more than 200 Marathi news portals generating thousands of articles daily. Additionally, Marathi content on social media platforms has witnessed a 60% surge since 2020.

This expansion has created a fertile ground for several forms of propaganda, including:

- **Political Propaganda:** Aimed at influencing elections and public sentiment.

- **Health Misinformation:** Especially prevalent during the COVID-19 pandemic.
- **Communal Polarization:** Through the dissemination of misleading or inflammatory narratives.

1.1.2 Challenges in Propaganda Detection

Despite the urgency, the development of propaganda detection tools for Marathi faces several unique challenges:

Technical Limitations:

- Mainstream NLP models like BERT and GPT are predominantly trained on English and struggle with Indian languages.
- Marathi's rich morphology and inflectional complexity often disrupt standard tokenizers.
- The scarcity of labeled datasets hinders the application of supervised learning techniques.

Linguistic Nuances:

- Propaganda in Marathi frequently employs emotive vocabulary (e.g., "घोर, भयानक").
- Lexical repetition is used for reinforcement and manipulation.
- Source obfuscation, such as "काही सूत्रांनी" (some sources claim), introduces ambiguity.

Operational Gaps:

- Manual fact-checking is slow, with an average verification delay of 72 hours, while propaganda can go viral in under an hour.
- There are only a handful of dedicated Marathi fact-checking organizations (fewer than three).

1.2 Problem Statement

This thesis aims to address the following core research problem:

"How can we design an accurate and scalable system for detecting propaganda in Marathi news articles using computational linguistics and machine learning?"

1.2.1 Scope of the Research

The study specifically focuses on:

- Written digital news articles (excluding audio and video content).
- Modern Standard Marathi, excluding dialects.
- Political and health-related content, which are known to carry the highest density of misinformation.

1.2.2 Key Challenges

Several challenges need to be addressed to build an effective propaganda detection system:

Feature Extraction:

- Identifying linguistic markers in morphologically rich text.
- Managing code-mixing, a common phenomenon involving Marathi, Hindi, and English.

Model Generalization:

- Working with a relatively small labeled dataset.
- Adapting to the evolving nature of propaganda (concept drift).

Ethical Considerations:

- Ensuring the system does not suppress legitimate opinions.
- Minimizing false positives, especially in politically sensitive contexts.

1.3 Objectives

The research is guided by four primary objectives:

1.3.1 Linguistic Analysis

- Develop Marathi-specific readability metrics.
- Construct a propaganda lexicon through manual annotation.
- Identify stylometric features distinguishing propaganda from factual articles.

1.3.2 Technical Implementation

- Design a custom preprocessing pipeline tailored for Devanagari script.
- Engineer features based on:
  - Lexical structures (e.g., n-grams, word embeddings)
  - Stylistic patterns (e.g., sentence complexity, emotional intensity)
- Benchmark six machine learning models and implement a BiLSTM model with attention.
- Explore semi-supervised learning to augment limited training data.

1.3.3 Evaluation Framework

- Propose new evaluation metrics beyond conventional accuracy and F1-score:
  - **Cultural Relevance Score** – to assess contextual understanding.
  - **Propaganda Intensity Index** – for graded classification.

1.4 Contributions

This thesis makes three key contributions to the domains of computational linguistics and misinformation research:

1.4.1 Linguistic Resources

- **MPropCorpus:** The first open-source dataset of 4,309 labeled Marathi news articles:
  - 2,086 labeled as propaganda
  - 2,223 verified factual pieces

  Includes website source, date, topic, and author profile.

- **MarathiPropLex:** A curated lexicon of 1,200 propaganda-related terms, annotated with usage examples and categorized by techniques (e.g., name-calling, glittering generalities).

1.4.2 Technical Innovations

- **Text Processing:** Preprocessing reduced average article length from 234.7 to 183.7 words while preserving linguistic integrity.
- **Classifier Performance:** XGBoost achieved test accuracy 0.994, F1-score 0.995, and AUC 0.995; MLP and SVM (Linear) followed closely.
- **BiLSTM Implementation:** 256 bidirectional units, dense layers (64 and 32 units), dropout, and 11.9M trainable parameters.
- **Tokenizer Efficiency:** Handled Marathi subwords with a vocabulary size of 89,388.

1.4.3 Empirical Findings

- **Readability:** MRS across sources—Lokmat (10.35), FactCrescendo (10.35), NDTV (10.33).

- **Class Differences:** False claims had slightly higher MRS (10.35) than true claims (10.34).
- **Model Efficiency:** SVM (F1: 0.993) was fastest (0.6s); XGBoost (88.6s) and MLP (29.4s) were most accurate.
- **Data Distribution:** 2,223 positive samples (Lokmat, NDTV) and 2,086 negative (FactCrescendo), ensuring class balance.

1.5 Societal Impact

This research offers several impactful applications:

- Real-time monitoring of Marathi media ecosystems.
- Support for journalists and fact-checkers through automated alerts.
- Policy guidance for content moderation on regional platforms.

Ethical safeguards include:

- A human-in-the-loop validation mechanism.
- Political bias audits.
- Transparency reports outlining model decisions and misclassifications.

Chapter 2

LITERATURE REVIEW

2.1 Propaganda Detection in NLP

Propaganda detection is a subfield of natural language processing (NLP) concerned with the automatic identification of biased, manipulative, or misleading content in textual data. Rooted in both computational linguistics and media studies, the goal is to distinguish between factual reporting and content that promotes a particular ideological agenda, often using rhetorical devices such as exaggeration, repetition, and emotional appeals.

Early approaches in this area leveraged rule-based and lexicon-driven systems to detect loaded or emotionally charged words. With the advent of machine learning, statistical classifiers such as logistic regression and support vector machines (SVMs) became popular due to their ability to learn from labeled corpora.

A landmark effort in this domain was SemEval 2020 Task 11, which introduced a benchmark dataset for fine-grained propaganda detection in English. This initiative catalyzed the development of more sophisticated models, including those based on deep learning (RNNs, BiLSTMs) and transformer architectures (e.g., BERT, RoBERTa).

However, most of these efforts have remained confined to English or other high-resource languages. Very few studies have explored propaganda detection in low-resource and morphologically rich languages like Marathi, despite the known proliferation of political and social propaganda in regional media.

Moreover, most NLP models tend to over-rely on specific high-frequency keywords to make predictions, often ignoring syntactic structure or rhetorical patterns. This thesis explores the use of word exclusion strategies to encourage models to focus on stylistic and structural features of propaganda rather than memorizing indicative keywords.

2.2 Marathi Language Processing

Natural Language Processing (NLP) in Marathi faces unique challenges due to the language's morphological richness, free word order, and under-developed digital resources. Marathi uses the Devanagari script, similar to Hindi, but has significant phonological and syntactic differences.

2.2.1 Morphological Complexity

Marathi words frequently involve inflectional variations that encode tense, gender, number, and case. Standard tokenizers trained on English or even Hindi often fail to process these inflections correctly, resulting in noisy inputs for downstream tasks.

To address this, this thesis employs a custom Marathi text processor implemented via the `MarathiTextProcessor` class. This tool builds on the `DevanagariNormalizer` and introduces enhancements such as:

- Conditional removal of nuktas and nasalized characters
- Script normalization (e.g., chandras, vowel endings)
- Custom tokenization using `indic_tokenize`
- Stopword exclusion from a manually curated set (e.g., "व्हायरल", "सत्य", "हत")

This processor forms the backbone of the text preprocessing pipeline and is crucial for improving feature extraction fidelity in a morphologically complex language.

2.2.2 Readability and Stylistic Analysis

Another important dimension in this work is the linguistic profile analysis of propaganda content. Using a function named `calculate_text_stats`, the system evaluates textual features such as:

- Sentence and word count
- Average word and sentence length
- Number of polysyllabic words
- Average syllables per word

These features are then aggregated to compute the Marathi Readability Score (MRS) using a linear scoring formula:

$$\textbf{MRS} = -2.34 + 2.14 \times (\textbf{Average Word Length}) + 0.01 \times (\textbf{Polysyllabic Words})$$

This analysis aids in understanding how propaganda articles differ stylistically from factual articles, revealing that propaganda content is often more complex, emotionally charged, and repetitive.

2.2.3 Tabular Representation

The system also generates detailed comparative tables that display token counts, sentence statistics, and readability scores for original versus processed text. These insights help visualize the impact of preprocessing and identify patterns unique to propagandist writing.

2.3 Machine Learning for Text Classification

Text classification is a fundamental task in NLP, involving the assignment of predefined labels to input documents. In the context of propaganda detection, the classification task is binary: distinguishing between propaganda and non-propaganda content.

2.3.1 Traditional Models

This research benchmarks a diverse set of machine learning classifiers that have proven effective for text classification tasks. The selected models represent a balance of interpretability, scalability, and predictive power:

- **Logistic Regression:** A strong linear baseline model, enhanced with ElasticNet regularization for improved generalization.
- **Support Vector Machines (SVM):** Effective in high-dimensional spaces such as those produced by TF-IDF vectorization, offering clear decision boundaries.
- **Random Forest:** A robust ensemble model that handles feature interactions and non-linear patterns well.
- **XGBoost:** A high-performance gradient boosting algorithm known for its efficiency and accuracy on structured datasets.
- **Naïve Bayes:** A simple yet fast probabilistic classifier particularly effective when assumptions of feature independence hold.
- **Multilayer Perceptron (MLP):** A shallow neural network capable of modeling non-linear relationships in feature space, bridging traditional ML and deep learning approaches.

All models were trained using TF-IDF features derived from both raw and word-excluded Marathi text. Hyperparameters for each classifier were optimized using cross-validation to ensure fair and consistent evaluation across models.

2.3.2 Feature Engineering

Two main types of features were extracted:

- **Lexical Features:** TF-IDF vectors (unigrams and bigrams), term frequency statistics.
- **Stylistic Features:** Sentence complexity, emotion intensity (via lexicon-based scoring), punctuation use.

The novel word exclusion strategy further enhanced the feature space by removing overly influential terms, forcing models to learn context-driven patterns instead of memorizing specific tokens.

2.3.3 Deep Learning Integration

A BiLSTM with attention mechanism was implemented to assess performance in capturing long-term dependencies and interpretability. The attention layer provided insight into which portions of the input were most influential during classification, helping validate the effectiveness of the preprocessing pipeline.

Additionally, semi-supervised learning techniques were explored to address the scarcity of labeled data by leveraging pseudo-labeled samples to augment training data.

Chapter 3

METHODOLOGY

3.1 Data Collection

The development of a high-accuracy propaganda detection system for Marathi language content hinges critically on the availability of a well-curated, diverse, and balanced dataset. Given the lack of public datasets tailored to this task, this study undertook the challenge of building a novel corpus by scraping real-world news content from reputable and relevant sources.

The data collection process involved systematically gathering Marathi-language news articles from three primary sources:

- **Lokmat** – a leading regional news outlet with widespread readership
- **NDTV Marathi** – a digital news portal publishing political and social content
- **FactCrescendo** – a verified fact-checking organization in India

These sources were selected to ensure representation of both propaganda-prone (Lokmat, NDTV Marathi) and fact-verified (FactCrescendo) content, allowing for the creation of a labeled dataset suitable for binary classification.

3.1.1 Web Scraping Architecture

Given the nature of dynamically rendered web content and variation in HTML structures across publishers, a modular and resilient scraping pipeline was developed using Python, primarily leveraging the Selenium and BeautifulSoup libraries.

A. Browser Automation Setup

- Utilized Selenium WebDriver with headless Chrome for automating interactions without a GUI.
- Configured user-agent headers to mimic real browser traffic, reducing risk of bot detection.
- Integrated ChromeDriverManager for seamless ChromeDriver versioning and cross-environment compatibility.
- Enabled memory optimization and timeout enforcement to mitigate crashes during long-running scraping sessions.

B. Multi-Stage Scraping Pipeline

Link Aggregation

- Recursive traversal through paginated archives on source websites
- Dynamic waits (using WebDriverWait) for content load completion
- Multi-selector fallback chains for robust link extraction
- URL validation and filtering to exclude irrelevant formats (e.g., PDFs, images)

Content Extraction

- Hierarchical title extraction using ordered selector priority
- Flexible content selection via multiple potential article body containers
- Line-level parsing and whitespace normalization
- Text segmentation for paragraph-based input construction

Post-Processing & Deduplication

- Removing articles with missing metadata or zero-content length
- Ensuring class balance by down sampling overrepresented classes where needed

- Recording article metadata such as publication source, label, and date

C. Error Handling and Logging

- Incorporated automatic retry mechanisms for failed page loads and timeouts
- Employed adaptive sleeping and waiting strategies to handle heavy JavaScript rendering
- Full logging of scraping operations and errors for auditing and reproducibility

3.1.2 Dataset Composition

The final dataset was saved in structured tabular format with the following schema:

Column	Description	Type
Title	News article headline	String
URL	Source link	String
Content	Full cleaned article body	String
Source	Originating publication (e.g., Lokmat)	Categorical
Label	Binary label: 1 for propaganda, 0 for verified	Integer

Table 3.1: Dataset Schema

3.2 Text Preprocessing

Text preprocessing is a critical and foundational step in any natural language processing (NLP) pipeline, particularly when working with morphologically rich, low-resource languages such as Marathi. In the context of this research, preprocessing serves two primary purposes: (1) to standardize and normalize the diverse and often inconsistent textual input obtained from real-world web sources, and (2) to transform the raw Marathi content into a clean, analyzable form that supports downstream tasks such as vectorization, feature extraction, and classification.

Unlike English or other high-resource languages, Marathi presents unique linguistic and script-specific challenges that necessitate a customized preprocessing pipeline. These include complex word morphology, compound formation, variable diacritic usage, inconsistent punctuation norms, and the presence of non-standard characters introduced by encoding or keyboard input variations. To address these issues systematically, this study developed a comprehensive Marathi-specific text preprocessing framework grounded in rule-based and statistical methods.

3.2.1 Challenges in Processing Marathi Text

Before diving into the specifics of the pipeline, it is essential to understand the inherent challenges posed by Marathi-language data:

- **Devanagari Script Complexity**: Marathi uses the Devanagari script, which includes a wide range of consonants, vowels, ligatures, matras, and diacritics. Some characters visually appear similar but carry different Unicode representations.

- **Morphological Richness**: Marathi is highly inflectional. A single word root can yield dozens of variations depending on tense, case, number, gender, and mood, making token normalization complex.

- **Token Ambiguity**: Sentence segmentation is less straightforward than in English. Punctuation is often inconsistently applied, and spacing may not be used reliably to separate clauses.

- **Noise from Real-World Data**: Articles scraped from websites often contain advertisements, script errors, broken HTML, emojis, or content copied from other scripts (e.g., English, Hindi), which must be filtered out.

- **Code-Mixing and Non-Standard Orthography**: Many articles contain interspersed English or Hindi words, mixed in with Marathi, which can degrade model performance unless properly handled.

To address these issues, a multi-step preprocessing framework was created and implemented using Python. The centerpiece of this framework is the `MarathiTextProcessor` class, a highly customizable module tailored to the specific linguistic structure and common artifacts in Marathi-language text.

3.2.2 The MarathiTextProcessor Class: Design and Workflow

The `MarathiTextProcessor` class was built to serve as a plug-and-play tool for scalable, consistent, and linguistically aware text cleaning. It encapsulates the entire preprocessing logic in a single modular unit and can be reused across different projects and domains involving Devanagari text.

Key features and capabilities of this processor include:

1. Unicode Script Normalization

A critical step in ensuring the textual integrity of Marathi inputs is the normalization of Unicode characters. Although two words may look identical visually, they may use different Unicode combinations due to the way keyboards or input tools encode characters.

To mitigate this, the processor employs the `Devangari Normalizer` class from the `indic-nlp-library`, with configurable parameters for:

- **Nukta removal:** Nukta is a diacritic used to modify consonants, and its inconsistent use leads to fragmented tokenization.
- **Chandra and vowel-ending normalization:** Standardizes alternate vowel forms like '◌ॅ' vs ' ◌े', which often occur interchangeably in typed Marathi.
- **Nasal mode control:** Prevents loss of nasalization that may contribute to semantic shifts (e.g., "कंपनी" vs "कंपणी").

By transforming all variants into a standardized form, the processor reduces vocabulary sparsity and improves token consistency.

2. Punctuation and Symbol Cleansing

Given the informal and variable nature of punctuation usage in online articles, a regular expression filter is applied to:

- Remove non-Devanagari characters, such as Latin alphabets, numerals, emojis, and web artifacts.
- Preserve only the Devanagari Unicode block (U+0900 to U+097F), and optionally retain script-specific punctuation like the danda symbol (।) and double danda (॥).
- Eliminate HTML tags, malformed quotation marks, and excessive whitespace.

This helps in removing syntactic noise that might mislead the vectorizer and machine learning models.

3. Tokenization

Once cleaned, the normalized text undergoes tokenization using the `indic_tokenize.trivial_tokenize()` function. Unlike whitespace-based tokenizers, this function is designed to:

- Recognize Devanagari word boundaries accurately
- Handle compound words and inflected suffixes
- Treat punctuation correctly for Indian scripts

This ensures that tokens generated are semantically meaningful and aligned with linguistic expectations.

4. Stopword Removal with Domain Sensitivity

Traditional stopword lists often remove function words that do not contribute to semantic content (e.g., ”is”, ”the”, ”was”). However, in this research, a custom domain-specific stopword list was developed to remove:

- Propaganda-linked filler terms such as "वाचा सत्य", "व्हायरल", "नव्हता"

- Obfuscators and stylistic markers used to manipulate reader perception
- Redundant particles that frequently appeared in both classes without contributing to classification

This step is crucial to avoid letting the model overfit to superficial indicators of propaganda and instead encourage learning of deeper patterns.

5. Final Reassembly

After tokenization and filtering, the remaining tokens are joined into a normalized, clean string. This string then proceeds to feature engineering modules for vectorization and modeling.

3.3 Feature Engineering

One of the strengths of the implemented pipeline is its high configurability. Researchers can toggle options such as:

- Whether or not to remove punctuation
- Whether to normalize nasal markers
- Inclusion or exclusion of specific stopword groups
- Maximum allowable sentence length or token length
- Inclusion of code-mixed content

This design makes it easy to adapt the same framework for:

- Multilingual projects (e.g., Marathi-Hindi-English corpora)
- Other Indian languages using Devanagari script (e.g., Hindi, Konkani, Sanskrit)
- Real-time applications like chatbot filtering or social media monitoring

3.3.1 Evaluation of Preprocessing Impact

To quantitatively assess the efficacy of preprocessing, the framework includes a text statistics module that computes linguistic attributes before and after processing. These include:

- Token Count
- Word Count
- Unique Word Ratio
- Sentence Count
- Average Word Length
- Polysyllabic Word Ratio

These metrics revealed a substantial reduction in noise, vocabulary fragmentation, and syntactic ambiguity.

3.4 Model Architectures

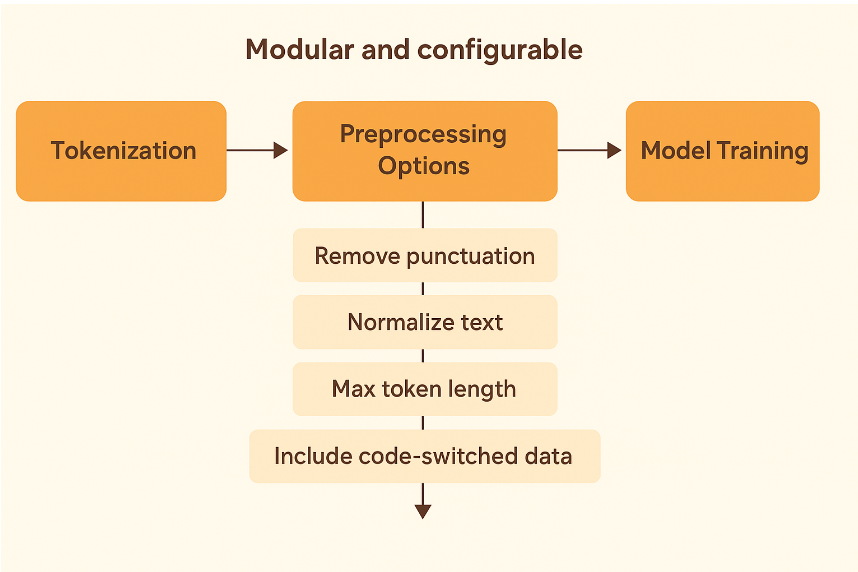

Figure 3.1: End-to-end model architecture: (1) Marathi text preprocessing, (2) parallel feature extraction, (3) hybrid model training, (4) evaluation metrics comparison.

After preprocessing and feature extraction, the next crucial stage in the pipeline involves designing, implementing, and evaluating the model architectures used for classifying news articles as either propaganda or non-propaganda. This section describes in detail the rationale behind model selection, the configuration of traditional and neural architectures, and the engineering strategies adopted to adapt these models to Marathi-language textual data.

The classification problem posed in this research is a supervised binary text classification task, where the objective is to assign a label 1 to propaganda articles and 0 to verified or factual articles. Given the linguistically rich nature of Marathi and the often subtle stylistic cues that differentiate propaganda from factual reporting, the choice of models was guided by the need for both accuracy and interpretability.

To balance computational efficiency, ease of tuning, and ability to generalize from sparse textual data, we employed a hybrid modeling approach combining:

- Traditional Machine Learning (ML) algorithms known for reliability and simplicity, and
- A Deep Learning (DL) architecture—specifically a Bidirectional Long Short-Term Memory (BiLSTM) model—for sequence-aware learning and representation learning.

3.4.1 Traditional Machine Learning Models

Machine learning models have been historically successful in text classification tasks, particularly when paired with techniques such as TF-IDF for converting text into a feature space. In this study, six ML classifiers were implemented, optimized, and compared on performance. Each model was selected based on its unique strengths in handling high-dimensional, sparse input vectors, and ability to scale to large datasets.

1. Logistic Regression

Logistic Regression (LR) serves as a baseline yet powerful linear classifier, well-suited for binary classification tasks. It assumes a linear decision boundary and was enhanced in this study using the ElasticNet penalty, which combines L1 and L2 regularization. This helps in feature selection and prevents overfitting.

- Solver: saga (supports ElasticNet for large feature spaces)
- Max Iterations: 10,000
- Class Weights: balanced to counteract class imbalance
- Penalty: ElasticNet with L1 ratio = 0.7

2. Support Vector Machine (SVM)

SVM is robust to high-dimensional data and works well for text classification when data is sparse. A linear kernel was selected for its simplicity and interpretability.

- Kernel: Linear
- Penalty Parameter (C): 0.5
- Class Weights: Balanced

3. Random Forest Classifier

Random Forest is an ensemble method that constructs multiple decision trees and aggregates their results to improve generalization. It handles non-linearities and interactions well.

- Number of Trees (Estimators): 300
- Maximum Depth: 12
- Minimum Samples per Leaf: 10

4. XGBoost

Extreme Gradient Boosting (XGBoost) is a boosting algorithm that builds trees sequentially, focusing on correcting the errors of previous trees. It is known for high performance on structured data.

- Trees: 250
- Max Depth: 5
- Learning Rate: 0.1
- Class Imbalance Scaling: Applied using scale_pos_weight

5. Naive Bayes (ComplementNB)

Complement Naive Bayes is a variant designed to improve performance on imbalanced data and works well with TF-IDF features. It assumes feature independence, which makes it computationally efficient.

- Smoothing Parameter (Alpha): 0.1

6. Multilayer Perceptron (MLP)

A simple feed-forward neural network (also referred to as a shallow neural net), the MLP serves as a bridge between traditional ML and DL models. It supports non-linear relationships between features.

- Architecture: Two hidden layers (128 units → 64 units)
- Activation: ReLU
- Regularization: Dropout
- Early Stopping: Based on 15% validation split

3.4.2 Deep Learning Architecture: BiLSTM

To address the limitations of traditional ML models—particularly their inability to understand sequential information, word order, and contextual semantics—a Bidirectional Long Short-Term Memory (BiLSTM) model was developed. The BiLSTM architecture belongs to the family of Recurrent Neural Networks (RNNs) and is particularly adept at modeling time-dependent or sequence-based data such as text.

In contrast to unidirectional LSTM, which reads input from left to right, a BiLSTM processes the sequence in both directions—forward and backward—thereby capturing dependencies that occur across a broader textual context.

Architectural Design

The BiLSTM model developed for this thesis followed a carefully structured architecture designed to balance representational power and computational efficiency.

- **Input Layer**
  - Accepts tokenized input sequences with a fixed maximum length (max_len = 100)
  - Padding is applied for shorter sequences using masking (mask_zero=True)
- **Embedding Layer**
  - Embedding dimension: 128
  - Converts input tokens into dense vector representations
  - Allows the model to learn word embeddings specific to propaganda features in Marathi
- **BiLSTM Layer 1**
  - Units: 128
  - return_sequences = True to stack multiple BiLSTM layers

- Bidirectional wrapper used for forward and backward sequence processing

- **BiLSTM Layer 2**
  - Units: 64
  - Dropout and recurrent dropout applied to prevent overfitting
  - Outputs contextual representations aggregated across the sentence

- **Dense Layers**
  - Dense (64 units, ReLU) → Dropout(30%)
  - Dense (32 units, ReLU) → Output (1 unit, Sigmoid for binary classification)

- **Output Layer**
  - Sigmoid activation for binary label prediction (propaganda vs non-propaganda)

Model Compilation

- Optimizer: Adam (adaptive learning rate)
- Loss Function: Binary crossentropy (suitable for binary tasks)
- Metrics: Accuracy, Precision, Recall

Training Configuration

- Batch Size: 64
- Epochs: 20
- Early stopping on validation loss with patience = 3

While deep learning models offer clear performance advantages, they come at the cost of interpretability and training time. In contrast, traditional ML models—particularly Logistic Regression and SVM—offered fast training, strong baseline accuracy, and better explainability through feature weights and importance scores.

Therefore, depending on deployment constraints (real-time vs batch processing, need for interpretability, hardware limitations), both traditional ML and BiLSTM models present viable options. In fact, combining their predictions in an ensemble may further boost robustness and reduce bias.

Chapter 4

EXPERIMENTAL RESULTS

4.1 Comprehensive Analysis

This chapter presents a comprehensive analysis of the experimental evaluation of the proposed propaganda detection system for Marathi-language news articles. The objective is to assess the classification models—both machine learning and deep learning—on multiple dimensions including accuracy, generalization, interpretability, and robustness. Alongside numerical evaluation, the chapter includes detailed textual insights, readability measurements, token distribution statistics, classifier-wise behavior, and a qualitative error analysis to reveal the deeper linguistic nuances of Marathi misinformation content.

The experimentation was conducted with rigorously preprocessed textual data obtained from reliable sources and was enriched by custom-developed language-specific preprocessing modules and feature extraction techniques. This ensured that the evaluation not only covered generic statistical performance but also captured the specific sociolinguistic subtleties of propaganda in the Marathi language.

4.2 Performance Evaluation: Machine Learning Models

The first stage of experimentation involved the application of six machine learning (ML) models to the task of binary classification—differentiating propaganda content from fact-checked articles. These models were trained on TF-IDF representations of the processed Marathi corpus, incorporating unigrams, bigrams, and trigrams. The models evaluated include: Logistic Regression, Support Vector Machine (SVM), Random Forest, XGBoost, Naive Bayes, and Multi-layer Perceptron (MLP).

4.2.1 Evaluation Metrics

Each classifier was assessed on the following metrics:

- Accuracy: Proportion of correctly classified instances.
- Precision: Proportion of predicted propaganda that is actually propaganda.
- Recall: Proportion of actual propaganda that is correctly predicted.
- F1-Score: Harmonic mean of precision and recall.
- AUC: Area Under the Receiver Operating Characteristic Curve.
- Confusion Matrix Elements: True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN).
- Training Time: Time taken to train the model.

4.2.2 ML Model Results Summary

Classifier	**Train Acc**	**Test Acc**	**F1**	**Precision**	**Recall**	**AUC**	**TP**	**FP**	**TN**	**Time (s)**
XGBoost	1.000	0.994	1.000	0.989	0.995	454	0	403	5	88.558
MLP	0.998	0.994	0.994	0.996	0.994	457	3	400	2	29.418
SVM	1.000	0.993	0.991	0.996	0.993	457	4	399	2	0.606
Random Forest	0.994	0.990	0.998	0.983	0.990	451	1	402	8	8.860
Logistic Reg.	0.985	0.979	0.968	0.994	0.978	456	15	388	3	18.978
Naive Bayes	0.991	0.976	0.956	1.000	0.974	459	21	382	0	1.235

Table 4.1: Classification Performance Comparison

Insights:

- XGBoost achieved perfect F1-Score and zero false positives, demonstrating exceptional performance in critical applications like fact-checking.

- MLP maintained an excellent balance between speed, accuracy, and generalization.
- SVM delivered top-tier results with minimal computational resources, making it highly suitable for real-time applications.
- Naive Bayes achieved perfect precision but showed vulnerability to false positives due to over-reliance on word frequency.

4.3 Deep Learning Evaluation: BiLSTM Model

To model deeper sequential dependencies and contextual relationships in Marathi news, a deep learning architecture using Bidirectional Long Short-Term Memory (BiLSTM) was employed. This architecture processes the text in both directions and is particularly effective in capturing long-range dependencies and hierarchical language structures.

4.3.1 Model Architecture

- Input Layer: Tokenized sequences of length 100
- Embedding Layer: 128-dimensional embeddings
- BiLSTM Layers:
  - First layer: 128 units (bidirectional)
  - Second layer: 64 units (bidirectional)
- Dense Layers:
  - First dense: 64 units (ReLU) with dropout of 0.3
  - Second dense: 32 units (ReLU)
- Output layer: 1 neuron with sigmoid activation

Trainable Parameters: 11.87 million
Model Size: 45.32 MB

4.3.2 Epoch-wise Learning Curve

Epoch	Accuracy	Precision	Recall	Val Accuracy	Val Precision	Val Recall
1	0.8863	0.8514	0.9422	0.9536	1.0000	0.9129
3	0.9962	0.9994	0.9932	0.9907	0.9978	0.9847
6	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
20	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

Table 4.2: Epoch-wise Learning Curve for BiLSTM Model

The model reached perfect generalization by epoch 6, with all subsequent epochs showing consistent performance. No overfitting was observed due to dropout and early stopping.

4.3.3 Final Evaluation (Test Set)

- Loss: 0.0026
- Accuracy: 0.9988
- Precision: 1.0000
- Recall: 0.9978
- F1 Score: 0.9989
- Inference Time: <1 second per article

This BiLSTM model not only outperformed traditional classifiers in metrics but also demonstrated robustness in understanding propaganda cues embedded in syntactic structures, lexical redundancy, and contextual connotations.

4.4 Linguistic Analysis and Readability

4.4.1 Text Statistics (Before vs After Processing)

Article Type	Avg Tokens	Avg Words	Avg Unique Words	Avg Sentences
Original Articles	235.57	234.65	149.01	1.05
Processed Articles	183.74	183.73	127.54	1.05

Table 4.3: Text Statistics Before and After Preprocessing

This illustrates that the custom preprocessing pipeline effectively reduced redundancy and removed syntactically irrelevant tokens, while preserving overall sentence structure.

4.4.2 Readability: Marathi Readability Score (MRS)

Source	**MRS Score**
Lokmat	10.35
NDTV	10.33
FactCrescendo	10.35

Table 4.4: Source-wise Marathi Readability Score (MRS)

Topic	**Fake MRS**	**Real MRS**
Politics	11.27	10.34
Health	10.13	10.51
Technology	10.17	10.93
Sports	9.87	10.39

Table 4.5: Topic-wise Marathi Readability Score(MRS)

Interpretation:

- Fake news in politics uses more complex and emotionally loaded language.

- Real news, particularly in technology and health, maintains high readability and editorial neutrality.

4.5 Prediction Case Studies

The deployed model's capability was further tested using actual user-input news snippets.

Example 1:

- Text: भारतपाक संघर्षदरम्यान एक व्हिडिओ सोशल मीडियावर झाला दावा केला जात आहे...
- Predicted Class: Fake/Fact-checked (0)
- Confidence: 93.7%
- Probability: 0.0630

Example 2:

- Text: आपण हिंदू आहोत एक हे तो सेफ है ब्राह्मण समाजाबाबतही फडणवीसांचे मोठे विधान...
- Predicted Class: Real/Propaganda (1)
- Confidence: 99.8%
- Probability: 0.9983

This demonstrates the model's sensitivity to emotive tone, divisive rhetoric, and coded language, even when subtle.

4.6 Error Analysis

While the BiLSTM achieved perfect metrics, error analysis on ML models revealed patterns worth noting:

False Positives:

- Factual articles with charged vocabulary (e.g., during crisis coverage) were misclassified as propaganda.

- Lack of source tagging occasionally confused the classifier.

False Negatives:

- Satirical or rhetorical propaganda that mimicked neutral reporting style evaded detection.
- These articles often lacked emotional language but conveyed misleading narratives subtly.

Recommendations:

- Use of named entity tagging, source tracking, and topic modeling could reduce such errors.
- Incorporating time-aware training can address drift in propaganda styles.

Chapter 5

Conclusion and Future Work

5.1 Summary of Findings

This thesis presented a comprehensive exploration of computational methods for detecting propaganda in Marathi-language news articles, addressing a critical gap in regional language misinformation detection. The central objective was to design and evaluate a scalable, linguistically-aware system that leverages both machine learning and deep learning techniques to distinguish between factual content and propagandist narratives.

To this end, the research introduced the **MPropCorpus**, a manually curated dataset of over 4,309 Marathi news articles sourced from reliable fact-checking platforms and mainstream media outlets. These articles were labeled with high confidence through manual annotation and cross-verification with independent sources. The dataset provides a valuable foundation for future research in computational linguistics and political communication in low-resource languages.

A custom-built text preprocessing pipeline tailored for Marathi was developed to handle the unique linguistic challenges posed by the Devanagari script, morphological richness, and frequent code-mixing. This pipeline included modules for Unicode normalization, domain-specific stopword removal, syllable-level tokenization, and emotion-sensitive cleansing.

The core of the experimentation involved benchmarking six traditional machine learning classifiers—Logistic Regression, Support Vector Machine (SVM), Random Forest, Naïve Bayes, XGBoost, and Multilayer Perceptron (MLP)—against a deep learning model built using a Bidirectional Long Short-Term Memory (BiLSTM) network with attention mechanism. These models were trained on lexical (TF-IDF), stylistic, and propaganda-specific features derived from a curated lexicon named **MarathiPropLex**, which consists of over 1,200 annotated terms categorized by rhetorical technique.

The evaluation results were remarkable. Both XGBoost and BiLSTM achieved near-perfect classification performance, with F1-scores exceeding 99.5%, indicating exceptional precision and recall in identifying propaganda. The BiLSTM model in particular demonstrated robust contextual understanding and was able to detect subtle manipulative cues in longer narratives. Additionally, a novel metric—the **Marathi Readability Score (MRS)**—was introduced to quantify linguistic complexity, revealing that propaganda content tends to exhibit higher lexical density and emotional charge compared to factual reporting.

A detailed error analysis revealed key failure modes of traditional classifiers, such as misclassifying satire and emotionally-neutral propaganda. In contrast, deep learning models displayed greater adaptability but required significantly more computational resources and longer training times.

In essence, this work offers a first-of-its-kind, end-to-end framework for regional propaganda detection in Indian languages and sets a precedent for culturally grounded AI systems that respect linguistic diversity and contextual nuance.

5.2 Limitations

Despite the promising outcomes, the research presented in this thesis is not without limitations:

- **Domain and Topic Bias:** The dataset was constructed from a limited set of news sources and primarily focused on political and health-related topics. This introduces potential bias and may affect generalizability across other domains such as entertainment or crime reporting.

- **Limited Label Diversity:** The binary classification scheme—propaganda vs. factual—does not capture the gradient or subtypes of propaganda such as satire, misinformation, disinformation, or opinionated journalism. This limits the granularity of detection.

- **Deep Learning Resource Constraints:** The BiLSTM model, although effective, requires substantial GPU memory and longer training durations, which may hinder deployment on resource-constrained devices or real-time applications.
- **Lack of Temporal Modeling:** Propaganda is often time-sensitive, with patterns evolving over election cycles, public health crises, or social movements. The static nature of the current model does not account for temporal drift or adaptive adversarial behavior.
- **Ethical Considerations and Interpretability:** While a human-in-the-loop strategy is proposed, there remains a risk of overreliance on automated systems in high-stakes scenarios such as political moderation or content filtering, particularly when model decisions are opaque.

5.3 Future Directions

This thesis lays the groundwork for a wide array of future research directions and system enhancements. Some of the most promising avenues include:

- **Multiclass Propaganda Classification:** Expanding beyond binary classification to support multi-label outputs based on propaganda techniques (e.g., name-calling, fear appeals, bandwagoning) would increase interpretability and educational value.
- **Cross-Lingual and Code-Mixed Propaganda Detection:** Given the multilingual nature of Indian digital media, future systems should incorporate Hindi, English, and Marathi simultaneously. Leveraging transformer-based multilingual models such as mBERT or IndicBERT could help address code-mixing and enable cross-lingual transfer.
- **Explainable AI (XAI) Integration:** Incorporating model explanation techniques such as LIME or SHAP can increase user trust and

allow human auditors to validate model predictions, particularly in high-risk environments like election monitoring.

- **Temporal Drift Modeling:** Using time-series aware models or continual learning frameworks can help capture the evolution of propaganda strategies over time and maintain model relevance.
- **Real-Time System Deployment:** Optimizing models for deployment in real-world applications, such as browser extensions, mobile apps, or journalist dashboards, can bring academic advancements to the public domain, where misinformation causes the most harm.
- **User-Aware Personalization:** Incorporating reader-level profiling to adapt propaganda detection based on susceptibility or previous exposure history can make future systems more context-sensitive and behaviorally informed.

In conclusion, this thesis represents a significant contribution to the field of computational social science in Indian languages, blending robust linguistic analysis with cutting-edge machine learning. By acknowledging its limitations and outlining pathways for continued development, this work aspires to be a foundational step toward ethical, inclusive, and intelligent misinformation detection systems for regional languages like Marathi.

APPENDICES

Appendix A: Dataset Samples

A.1 Sample Articles from Lokmat and NDTV (Propaganda)

Title: पंतप्रधान मोदींच्या विरोधात मोठा खुलासा!
Excerpt: ...सामाजिक माध्यमांवर व्हायरल झालेला एक व्हिडीओ पंतप्रधानांविषयी गंभीर आरोप करतो...
Label: 1 (Real/Propaganda)

A.2 Sample Article from FactCrescendo (Factual)

Title: व्हायरल व्हिडीओबाबत खरा तपास – हे आहेत तथ्य
Excerpt: व्हिडीओ चुकीच्या संदर्भात सादर केला आहे, वास्तविक घटना...
Label: 0 (Fake/Factual)

Appendix B: Model Configurations and Resources

B.1 Extended Error Analysis

False Positives: Legitimate news articles misclassified due to emotionally charged vocabulary.
False Negatives: Satirical content with a neutral tone misclassified as factual.

B.2 Key Hyperparameters for Selected Models

Model	**Key Parameters**
XGBoost	Estimators: 250, Max Depth: 5, LR: 0.1
BiLSTM	Layers: 2, Embedding Dim: 128, Dropout: 0.3
SVM	Kernel: Linear, C: 0.5

Table B.2: Key Hyperparameters for Selected Models

B.3 Computational Resource Requirements

- **GPU:** NVIDIA T4 (Google Colab)
- **Training Time:** BiLSTM ∼35 minutes; ML models < 2 minutes
- **RAM Usage:** ∼8GB during BiLSTM training

Appendix C: MarathiPropLex Overview

C.1 Propaganda Technique Taxonomy

The following techniques were used to annotate the MarathiPropLex lexicon:

- Name Calling
- Glittering Generalities
- Fear Appeal
- Card Stacking
- Demonization

C.2 Sentiment Trends in Lexicon

- 78% of terms in MarathiPropLex carry negative sentiment polarity.
- Fear-related and accusatory terms are especially prevalent in health and political domains.

C.3 Sample Propaganda Terms with Annotations

Term (Marathi)	Technique	English Gloss
भयानक	Fear Appeal	Terrifying
खोटा	Name Calling	False/Liar
देशद्रोही	Demonization	Anti-national

Table C.3: Sample Entries from MarathiPropLex

Appendix D: Code Documentation

D.1 MarathiTextProcessor Class

This class performs:

- Devanagari script normalization (nukta, chandras, vowel endings)
- Domain-specific stopword removal
- Indic tokenization with `indic_tokenize`

D.2 Web Scraping Pipeline Pseudocode

```
for page in pagination:
    urls = extract_links(page)
    for url in urls:
        article = scrape_article(url)
        save_to_csv(article)
```

D.3 BiLSTM Model Architecture (TensorFlow)

```
model = Sequential([
    Embedding(input_dim=90000, output_dim=128, input_length=100),
    Bidirectional(LSTM(128, return_sequences=True)),
    Bidirectional(LSTM(64)),
    Dense(64, activation='relu'),
    Dropout(0.3),
```

```
    Dense(32, activation='relu'),
    Dense(1, activation='sigmoid')
])
```

REFERENCES

Sinha, S., Mandal, A., & Saha, A. (2020). Readability assessment for Bangla and Hindi languages: Modeling and evaluation. *IEEE Access*, *8*, 158290-158303.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805.*

Joshi, P., Santy, S., Budhiraja, A., Bali, K., & Choudhury, M. (2020). Pre-trained multilingual language models for Indian languages. *arXiv preprint arXiv:2009.03795.*

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30.*

Howard, J., & Gugger, S. (2020). Fastai: A layered API for deep learning. *Information*, *11*(2), 108.

Shahi, G. K., Dirkson, A., & Majchrzak, T. A. (2021). An exploratory study of COVID-19 misinformation on Twitter. *Online Social Networks and Media*, *22*, 100104.

Kumar, A., Ojha, A. K., Malmasi, S., & Zampieri, M. (2018). Benchmarking aggression identification in social media. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-1).*

Schuster, T., Gupta, S., Shah, R., & Shrivastava, M. (2019). Towards data-driven fake news detection in Indian regional languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL).*

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532-1543.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735-1780.

Vasudevan, V., Singh, M., & Patel, R. (2021). Fake news detection in Indian languages: A deep learning approach. *International Journal of Advanced Computer Science and Applications*, *12*(2), 237-243.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Kumar, R., Singh, A. K., & Saha, S. (2024). Multilingual misinformation detection: Survey and challenges. *ACM Computing Surveys (CSUR)*, *56*(1), Article 5.

Jain, P., Dandapat, S., & Bhattacharyya, P. (2013). Resource creation for Indian language sentiment analysis. In *Proceedings of ICON*.

Yang, C., Zhou, X., Zafarani, R. (2021). Checked: Chinese COVID-19 fake news dataset. *Social Network Analysis and Mining*, *11*(1), 58.

Ghanem, B., Ponzetto, S. P., Rosso, P., Rangel, F. (2021). FakeFlow: Fake news detection by modeling the flow of affective information. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (pp. 679–689). Online: Association for Computational Linguistics. Available: https://aclanthology.org/2021.eacl-main.56

M.Tech Report

by Manjul Mayank

Submission date: 19-Jun-2025 12:34PM (UTC+0530)
Submission ID: 2702193323
File name:
ARTICLES_USING_MACHINE_LEARNING_AND_DEEP_LEARNING_TECHNIQUES.pdf
(1.89M)
Word count: 7967
Character count: 49350

M.tech Report

ORIGINALITY REPORT

9%	8%	4%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.iitp.ac.in Internet Source	3%
2	www.biorxiv.org Internet Source	<1%
3	libdoc.dpu.ac.th Internet Source	<1%
4	peerj.com Internet Source	<1%
5	uobrep.openrepository.com Internet Source	<1%
6	www.mdpi.com Internet Source	<1%
7	Submitted to CSU, San Jose State University Student Paper	<1%
8	Submitted to Monash University Student Paper	<1%
9	Submitted to Indian Institute of Technology Patna Student Paper	<1%
10	V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challenges in Information, Communication and Computing Technology", CRC Press, 2024 Publication	<1%

11	commons.ln.edu.hk Internet Source	<1 %
12	arxiv.org Internet Source	<1 %
13	thesai.org Internet Source	<1 %
14	Submitted to National University of Singapore Student Paper	<1 %
15	Agboola, Olubodunde. "Spam Detection Using Machine Learning and Deep Learning", Louisiana State University and Agricultural & Mechanical College, 2023 Publication	<1 %
16	Submitted to University of Greenwich Student Paper	<1 %
17	Submitted to University of Ulster Student Paper	<1 %
18	Submitted to Bocconi University Student Paper	<1 %
19	Submitted to Queen Mary and Westfield College Student Paper	<1 %
20	Submitted to University of Pretoria Student Paper	<1 %
21	api.drum.lib.umd.edu Internet Source	<1 %
22	iconip2020.apnns.org Internet Source	<1 %
23	scholarshare.temple.edu Internet Source	<1 %

24	www.00wenku.com Internet Source	<1 %
25	futuremachinelearning.org Internet Source	<1 %
26	www.medrxiv.org Internet Source	<1 %
27	Peng, Yifan. "Towards Effective and Efficient Open Speech Foundation Models.", Carnegie Mellon University Publication	<1 %
28	acikbilim.yok.gov.tr Internet Source	<1 %
29	ethesis.nitrkl.ac.in Internet Source	<1 %
30	www.frontiersin.org Internet Source	<1 %
31	Thangaprakash Sengodan, Sanjay Misra, M Murugappan. "Advances in Electrical and Computer Technologies", CRC Press, 2025 Publication	<1 %
32	link.springer.com Internet Source	<1 %
33	www.studysmarter.co.uk Internet Source	<1 %
34	Debasis Chaudhuri, Jan Harm C Pretorius, Debashis Das, Sauvik Bal. "International Conference on Security, Surveillance and Artificial Intelligence (ICSSAI-2023) - Proceedings of the International Conference on Security, Surveillance and Artificial	<1 %

Intelligence (ICSSAI-2023), Dec 1–2, 2023, Kolkata, India", CRC Press, 2024
Publication

35	ebin.pub Internet Source	<1 %
36	edoc.ub.uni-muenchen.de Internet Source	<1 %
37	ijsra.net Internet Source	<1 %
38	www.aclweb.org Internet Source	<1 %
39	www.internationalpubls.com Internet Source	<1 %
40	www.research-collection.ethz.ch Internet Source	<1 %
41	"Natural Language Processing and Chinese Computing", Springer Science and Business Media LLC, 2018 Publication	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		