****

**DS7010 : MSc DISSERTATION**

**Fake News Detection Using NLP And Machine Learning**

SUBMITTED TO: DR. YANG LI

SUBMITED BY: KANCHAN KUKRETI

U2642385

MSc DATA SCIENCE

Table of Contents

[ABSTRACT 4](#_Toc197131458)

[1 Introduction 5](#_Toc197131459)

[1.1 Research Problem and Motivation 6](#_Toc197131460)

[1.2 Research Objectives 7](#_Toc197131461)

[1.3 Research Significance 7](#_Toc197131462)

[1.4 Dissertation Structure 7](#_Toc197131463)

[2 Literature Review Overview 8](#_Toc197131464)

[2.1 Characteristics and Challenges of Fake News 8](#_Toc197131465)

[2.1.1 Common Traits of Fake News 8](#_Toc197131466)

[2.1.2 Detection Challenges 8](#_Toc197131467)

[2.2 Role of Natural Language Processing in Fake News Detection 9](#_Toc197131468)

[2.3 Traditional Machine Learning Models for Fake News Detection 9](#_Toc197131469)

[2.4 Feature Engineering Techniques 10](#_Toc197131470)

[2.5 Deep Learning Models for Fake News Detection 10](#_Toc197131471)

[2.6 Benchmark Datasets in Fake News Research 11](#_Toc197131472)

[2.7 Research Gaps in Existing Literature 11](#_Toc197131473)

[2.8 Summary 11](#_Toc197131474)

[3 Methodology Overview 12](#_Toc197131475)

[3.1 Dataset Description 12](#_Toc197131476)

[3.2 Data Pre-processing Pipeline 12](#_Toc197131477)

[3.3 Feature Engineering and Representation 13](#_Toc197131478)

[3.4 Model Development Pipeline 13](#_Toc197131479)

[3.5 Model Evaluation Strategy 14](#_Toc197131480)

[3.6 Hyperparameter Tuning 14](#_Toc197131481)

[3.7 Ethical Considerations 14](#_Toc197131482)

[3.8 Summary 15](#_Toc197131483)

[4 EDA Overview 16](#_Toc197131484)

[4.1 Dataset Labelling and Integration 16](#_Toc197131485)

[4.2 Duplicate Detection and Class Balance 16](#_Toc197131486)

[4.3 Exploratory Textual Analysis 16](#_Toc197131487)

[4.3.1 Basic Textual Feature Engineering 16](#_Toc197131488)

[4.3.2 Visual Analysis of Text Length 17](#_Toc197131489)

[4.3.3 Outlier Detection in Text Length 17](#_Toc197131490)

[4.3.4 Comparative Length Analysis by Class 17](#_Toc197131491)

[4.3.5 Manual Review of Outlier Articles 17](#_Toc197131492)

[4.3.6 Removal of Empty or Invalid Articles 17](#_Toc197131493)

[4.3.7 Final Preprocessing Visual Review 17](#_Toc197131494)

[4.4 Text Cleaning and Normalisation Pipeline 17](#_Toc197131495)

[4.4.1 Post-Cleaning Quality Assurance 18](#_Toc197131496)

[4.4.2 Post-Cleaning Class Verification 18](#_Toc197131497)

[4.5 Lemmatisation and Stopword Removal 18](#_Toc197131498)

[4.6 Engineered Linguistic Features 18](#_Toc197131499)

[4.6.1 Stylistic Feature Engineering 18](#_Toc197131500)

[4.6.2 Readability Metric Engineering 18](#_Toc197131501)

[4.7 Word Cloud and Keyword Frequency Analysis 19](#_Toc197131502)

[4.8 Sentiment Analysis using VADER 21](#_Toc197131503)

[4.9 Multivariate Feature Relationship Analysis: Pair Plot 22](#_Toc197131504)

[4.10 Named Entity Recognition (NER) for Semantic Insight 22](#_Toc197131505)

[4.11 Part-of-Speech (POS) Tagging 23](#_Toc197131506)

[4.12 Feature Correlation Analysis 25](#_Toc197131507)

[4.13 Keyword Co-occurrence and Network Analysis 26](#_Toc197131508)

[4.13.1 Bigram Analysis 26](#_Toc197131509)

[4.13.2 Network Graph Analysis 26](#_Toc197131510)

[4.14 Lexical & Semantic Complexity: Lexical Diversity 28](#_Toc197131511)

[4.15 Data Leakage Check & Class Imbalance Analysis 28](#_Toc197131512)

[4.16 Bias in Metadata: Subject Analysis 29](#_Toc197131513)

[4.17 Pairwise Relationships Between Features 30](#_Toc197131514)

[4.18 Dimensionality Reduction and Clustering 31](#_Toc197131515)

[4.18.1 PCA Clustering 31](#_Toc197131516)

[4.18.2 t-SNE Clustering 32](#_Toc197131517)

[4.18.3 UMAP Clustering 33](#_Toc197131518)

[4.19 Revisiting Readability Scores Post-UMAP 34](#_Toc197131519)

[4.20 Investigating Readability Score Outliers 34](#_Toc197131520)

[4.21 Readability Score Analysis and Visualisation 34](#_Toc197131521)

[4.22 Temporal Analysis: Preparing for Time-Based Trend Exploration 36](#_Toc197131522)

[4.22.1 Temporal and Content Analysis 36](#_Toc197131523)

[4.23 Feature Selection & Final Dataset Preparation 36](#_Toc197131524)

[4.24 Feature Engineering: Converting Text into Vectors 37](#_Toc197131525)

[4.24.1 TF-IDF Vectorisation 37](#_Toc197131526)

[4.24.2 GloVe Word Embeddings (SpaCy) 37](#_Toc197131527)

[5 Experiments and Results Overview 38](#_Toc197131528)

[5.1 Logistic Regression Evaluation 38](#_Toc197131529)

[5.2 Random Forest Evaluation 39](#_Toc197131530)

[5.3 Support Vector Machine (SVM) Evaluation 40](#_Toc197131531)

[5.4 XGBoost Evaluation 41](#_Toc197131532)

[5.5 LSTM Model Evaluation 42](#_Toc197131533)

[5.6 BERT Model Evaluation 44](#_Toc197131534)

[5.7 Model Validation Strategy 45](#_Toc197131535)

[5.7.1 Logistic Regression Validation 45](#_Toc197131536)

[5.7.2 Random Forest Validation 47](#_Toc197131537)

[5.7.3 Support Vector Machine (SVM) Validation 49](#_Toc197131538)

[5.7.4 XGBoost Classifier Validation 52](#_Toc197131539)

[5.8 Hyperparameter Tuning 54](#_Toc197131540)

[*5.8.1* Random Forest Classifier Hyperparameter Tuning 54](#_Toc197131541)

[*5.8.2* Support Vector Machine (SVM) Hyperparameter Tuning 54](#_Toc197131542)

[5.8.3 XGBoost Hyperparameter Tuning 55](#_Toc197131543)

[5.9 Validation Results of Tuned Models 55](#_Toc197131544)

[*5.9.1* Tuned Random Forest Classifier 55](#_Toc197131545)

[*5.9.2* Tuned Support Vector Machine (SVM) 56](#_Toc197131546)

[*5.9.3* Tuned XGBoost Classifier 57](#_Toc197131547)

[5.9.4 LSTM Model Performance 57](#_Toc197131548)

[5.10 Models Comparison 60](#_Toc197131549)

[6 Discussion : Interpretation of Model Performance 63](#_Toc197131550)

[6.1 Feature Engineering Insights 63](#_Toc197131551)

[6.2 Dimensionality Reduction and Visual Clustering 64](#_Toc197131552)

[6.3 Overfitting and Generalisation 64](#_Toc197131553)

[6.4 Class Imbalance and Bias Mitigation 64](#_Toc197131554)

[6.5 Contribution to Existing Literature 64](#_Toc197131555)

[6.6 Practical Implications 65](#_Toc197131556)

[6.7 Limitations 65](#_Toc197131557)

[7 Future Work 66](#_Toc197131558)

[*7.1* Integration of Multimodal Features 66](#_Toc197131559)

[*7.2* Temporal Dynamics and Event-Aware Detection 66](#_Toc197131560)

[*7.3* Cross-Domain and Cross-Language Generalisation 66](#_Toc197131561)

[*7.4* Explainable AI (XAI) for Trustworthy Decisions 66](#_Toc197131562)

[*7.5* Real-Time Implementation and Scalability 66](#_Toc197131563)

[*7.6* User Behaviour and Propagation Analysis 67](#_Toc197131564)

[7.7 Conclusion 67](#_Toc197131565)

[8 References 68](#_Toc197131566)

[Appendix: 69](#_Toc197131567)

# ABSTRACT

The rapid proliferation of fake news on digital platforms presents a significant threat to public trust, political stability, and information integrity. This dissertation investigates the development of an automated fake news detection system using Natural Language Processing (NLP) and Machine Learning (ML) techniques. The study employs a labelled dataset of over 44,000 news articles from Kaggle, combining extensive text preprocessing, feature engineering, and exploratory data analysis to uncover linguistic, semantic, and stylistic patterns distinctive to fake and real news.

Multiple models were developed and evaluated, ranging from traditional classifiers—Logistic Regression, Random Forest, SVM, and XGBoost—to deep learning architectures like LSTM and BERT. Feature engineering incorporated sentiment scores, readability metrics, named entity recognition, lexical diversity, and bigram analysis. Dimensionality reduction techniques such as PCA, t-SNE, and UMAP were used to explore class separability. Evaluation metrics included accuracy, F1-score, ROC-AUC, PR-AUC, and k-Fold Cross-Validation.

Results demonstrated that the BERT model significantly outperformed all others, achieving 99.94% accuracy and an F1-score of 1.00, highlighting the effectiveness of transformer-based architectures in capturing nuanced language patterns. The dissertation also addresses ethical concerns such as model interpretability and bias. The findings contribute to the existing body of research and offer a scalable, interpretable pipeline suitable for real-world misinformation detection.

*Keywords* : *BERT, Sentiment Analysis, Readability Metrics, Feature Engineering, Text Classification, Transformer Models*

**CHAPTER 1: INTRODUCTION**

# Introduction

The digital information landscape has undergone a profound transformation in recent years, driven by the rapid growth of online media and social platforms. While this evolution has democratised information access, it has also enabled the widespread dissemination of fake news—intentionally misleading or fabricated content that mimics legitimate journalism. The consequences of such misinformation are far-reaching, from undermining democratic processes and fuelling public health crises to eroding trust in institutions and media. During global events such as the COVID-19 pandemic, the impact of fake news became alarmingly evident, as misleading narratives spread faster than factual reporting (Brennen, et al., 2020). These developments underscore the urgent need for robust, scalable solutions to detect and mitigate the spread of misinformation.

Artificial Intelligence (AI), and more specifically Natural Language Processing (NLP) and Machine Learning (ML), offers promising avenues to automate the identification of fake news. While various models have been developed to tackle this challenge, the task remains complex due to linguistic subtleties, intentional manipulation, and the ethical implications of automated decision-making. This dissertation investigates these challenges and aims to develop an effective, interpretable, and ethical fake news detection system using a comparative approach across multiple machine learning and deep learning models.

## Research Problem and Motivation

Fake news is not simply a nuisance; it poses a serious threat to societal functioning, public safety, and democratic integrity. From politically motivated disinformation that sways elections to sensationalist claims that cause public panic, fake news continues to evolve in form and impact. The urgency to counter this problem has motivated researchers to explore automated detection techniques that can scale with the volume of content being produced online.

However, several technical and ethical challenges persist:

* ***Linguistic ambiguity*** and stylistic mimicry often blur the line between real and fake content.
* ***Effective feature extraction*** is vital for capturing relevant patterns in text but is non-trivial due to the varied nature of fake news.
* ***Model reliability and fairness*** are essential to avoid biased outcomes and ensure public trust.
* ***Deep learning models***, though powerful, often lack interpretability and require substantial computational resources.

This research addresses these concerns through a systematic evaluation of multiple modelling strategies, including both traditional and modern AI techniques.

## Research Objectives

This study aims to design and evaluate a robust fake news detection framework using advanced NLP and ML techniques. The primary objectives are:

* To investigate and understand the ***linguistic, structural, and stylistic features*** that characterise fake and true news content.
* To apply and compare ***a range of machine learning and deep learning models***, including Logistic Regression, Random Forest, SVM, XGBoost, LSTM, and BERT, for binary classification of news articles.
* To develop ***feature engineering strategies*** - including sentiment analysis, readability metrics, and named entity recognition - to enhance model performance.
* To critically assess ***model performance, scalability, and generalisation*** through cross-validation, learning curves, and standard evaluation metrics.
* To address ***ethical considerations*** such as algorithmic bias, interpretability, and the societal implications of automated misinformation detection.
* To propose a ***reproducible pipeline*** that can serve as a foundational framework for future research or real-world deployment.

## Research Significance

The outcomes of this research have both academic and practical relevance. By evaluating a diverse range of NLP and ML approaches, this study contributes to the growing body of literature on misinformation detection. The insights derived from feature analysis, model performance, and ethical considerations offer guidance to developers, journalists, and policymakers working to combat fake news. Furthermore, the proposed methodology provides a structured foundation that can be adapted to multilingual, multimodal, or real-time detection scenarios.

## Dissertation Structure

This dissertation is organised into the following chapters:

* ***Chapter 1 – Introduction***: Outlines the background, motivation, research problem, aims, and objectives of the study, along with a summary of the dissertation’s structure.
* ***Chapter 2 – Literature Review***: Provides a critical review of existing research on fake news detection, machine learning and NLP approaches, key methodologies, and identified research gaps.
* ***Chapter 3 – Methodology***: Describes the dataset used, preprocessing steps, feature engineering strategies, and the modelling framework. It also explains the rationale behind chosen algorithms and tools.
* ***Chapter 4 – Data Understanding and Exploration*:** Presents detailed exploratory data analysis (EDA), including lexical, sentiment, syntactic, and temporal investigations, along with dimensionality reduction and visual clustering.
* ***Chapter 5 – Model Development and Evaluation***: Discusses the training, tuning, and validation of multiple ML and deep learning models. It compares model performances using accuracy, F1-score, ROC-AUC, PR-AUC, and cross-validation metrics.
* ***Chapter 6 – Discussion and Interpretation***: Interprets key findings, evaluates feature importance, addresses ethical implications, and discusses the model's strengths and limitations.
* ***Chapter 7 – Conclusion and Future Work***: Summarises the study's contributions, limitations, and proposes directions for future research and potential real-world implementation.

**CHAPTER 2 : LITERATURE REVIEW**

# Literature Review Overview

The proliferation of fake news has emerged as a critical challenge in the digital era, with widespread implications for democracy, public health, and societal trust. The manual verification of news is infeasible at scale, leading to the exploration of automated fake news detection using Artificial Intelligence (AI) techniques, particularly **Natural Language Processing (NLP)** and **Machine Learning (ML).** While early approaches relied on superficial textual cues, recent studies emphasise the importance of understanding linguistic structure, sentiment, and contextual semantics to distinguish between genuine and fabricated content. This chapter reviews the characteristics and challenges of fake news, the evolution of NLP and ML-based detection models, benchmark datasets, and key research gaps.

## Characteristics and Challenges of Fake News

### Common Traits of Fake News

Fake news typically exhibits certain distinguishing features that deviate from professional journalism. These include:

* ***Manipulative language***: Often emotionally charged or sensationalist to provoke reaction.
* ***Lack of credible sourcing***: Frequently includes anonymous or unverifiable sources.
* ***High virality***: Designed for rapid online dissemination, exploiting social media dynamics.

Brennen et al. (2020) emphasised how fake news narratives during COVID-19 lacked grounding in evidence and were often designed to exploit public fear (Brennen, et al., 2020).

### Detection Challenges

Fake news detection is complicated by several key challenges:

* ***Linguistic ambiguity***: Fabricated content often mimics real news stylistically and structurally, blurring detection boundaries (Zhou & Zafarani, 2020).
* ***Subtle misinformation***: Articles may distort truth without fully falsifying facts, making binary classification difficult (Bender, et al., 2021).
* ***Algorithmic bias***: Models trained on biased datasets risk perpetuating unfairness (Shu, et al., 2017).
* ***Feature complexity***: Extracting meaningful patterns from high-dimensional, sparse text data remains non-trivial.

## Role of Natural Language Processing in Fake News Detection

**NLP** plays a foundational role in fake news detection by transforming raw text into structured features that ML algorithms can understand. These transformations include tokenisation, part-of-speech tagging, lemmatisation, sentiment analysis, and vector embeddings.

According to Raza and Ding (2022), traditional ML models such as Support Vector Machines and Random Forests rely heavily on manually engineered NLP features like TF-IDF and n-grams. In contrast, deep learning approaches (e.g. LSTM, BERT) benefit from rich contextual embeddings, allowing them to capture nuanced semantic relationships and discourse structures (Raza & Ding, 2022).

Moreover, advanced NLP techniques such as ***Named Entity Recognition (NER)*** and ***readability analysis***help capture the credibility and style of news articles, as fake news often manipulates entities or uses overly simplistic language (Shu, et al., 2017). These linguistic features are essential for improving classification performance.

## Traditional Machine Learning Models for Fake News Detection

Table 2.3.1 summarises the strengths and limitations of traditional ML classifiers along with commonly cited references.

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Strengths | Limitations | Citation |
| Logistic Regression | Simple, interpretable, good with sparse features (TF-IDF) | Limited ability to capture contextual information | (Shu, et al., 2017) |
| Support Vector Machine (SVM) | Effective in high-dimensional space | Computationally intensive, kernel selection required | (Zhou & Zafarani, 2020) |
| Random Forest | Robust to overfitting, handles feature importance well | Slow with large datasets, less effective with sequential data | (Lazer, et al., 2018) |
| XGBoost | High accuracy, good for structured data | Overfitting if not tuned, less explainable | (Raza & Ding, 2022) |

Table 2.3.1 Comparison of Traditional ML Algorithms for Fake News Detection

These algorithms rely on structured feature vectors derived from NLP techniques like bag-of-words, TF-IDF, and sentiment polarity. While interpretable and efficient, they struggle with context, irony, and ambiguity—key components in fake news text.

## ****Feature Engineering Techniques****

Effective text representation is critical to model performance in fake news detection. Common NLP-based feature extraction methods include:

* ***Bag-of-Words (BoW) and TF-IDF*:** Transform text into numeric vectors based on word frequency and importance; simple but context-agnostic.
* ***Sentiment Analysis*:** Captures emotional tone—often exaggerated in fake news (Vosoughi, et al., 2018).
* ***Readability Metrics*:** Fake news tends to have simpler syntax and excessive punctuation; metrics like Flesch and Gunning Fog quantify this (Shu, et al., 2017).
* ***Named Entity Recognition (NER):*** Identifies key entities (e.g., people, places) to assess content credibility.
* ***Word Embeddings*:** Pre-trained vectors like Word2Vec, GloVe, and BERT embed semantic relationships, outperforming basic frequency-based features.

While traditional models rely on these features, they lack the capacity to learn deeper linguistic structures—highlighting the need for deep learning approaches.

## Deep Learning Models for Fake News Detection

* *Recurrent Neural Networks (LSTM)* : Long Short-Term Memory networks (LSTM) are widely adopted for their ability to model long-range dependencies in text. They are especially effective in capturing the narrative flow and rhetorical structures in articles. However, they require large training data and are prone to overfitting (Devlin, et al., 2018).
* *Transformer-Based Architectures (BERT)* : Bidirectional Encoder Representations from Transformers (BERT) revolutionised NLP by allowing deep contextual understanding of language. BERT’s attention mechanism considers both forward and backward context, enabling superior performance in text classification tasks (Devlin, et al., 2018).

BERT has been used in recent studies such as Kaliyar et al. (2021), achieving state-of-the-art accuracy in fake news detection (Kaliyar, et al., 2021) . However, it requires fine-tuning and substantial computational resources.

* *Semantic Embeddings* : Techniques like Word2Vec and GloVe convert words into dense vector representations that reflect semantic similarity. While they outperform frequency-based models, they lack dynamic contextual awareness (Lazer, et al., 2018). In contrast, BERT embeddings are context-aware, making them ideal for fake news detection.

## Benchmark Datasets in Fake News Research

Several publicly available datasets have supported fake news research:

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | Description | Size | Reference |
| LIAR | Fact-checked political statements | 12,800 | (Wang, 2017) |
| Kaggle Fake News | Balanced real and fake news | 44,000 | Kaggle (2020) |
| Fake News Corpus | Large corpus of mixed articles | 1M+ | (Golbeck, et al., 2018) |
| COVID-19 Fake News | Health-related misinformation | ~10,000 | (Sharma, et al., 2020) |

Despite their usefulness, most datasets are limited to English and do not include multimedia content or social context, which restricts real-world generalisability.

## Research Gaps in Existing Literature

While NLP and ML have advanced the field of fake news detection, several gaps remain:

* *Limited Multilingual Capability*: Most detection models are trained on English datasets, limiting their effectiveness in non-English or cross-lingual contexts.
* *Lack of Interpretability in Deep Models*: Deep learning approaches often function as black boxes, offering high accuracy but little insight into how predictions are made.
* *Underuse of Hybrid Models*: Few studies combine textual features with network or user-based signals, missing opportunities for more holistic detection.
* *Neglect of Temporal Dynamics*: Static models do not account for the evolving nature of misinformation across time, events, or news cycles.
* *Ethical and Fairness Challenges*: Algorithmic bias, fairness, and risks of false positives require deeper attention to ensure responsible deployment (Bender, et al., 2021).

## Summary

This chapter reviewed the linguistic traits and detection challenges of fake news, the evolution of NLP-driven ML and DL approaches, and the datasets used in prior studies. It was observed that while traditional ML models offer interpretability, they fall short in semantic understanding. In contrast, advanced deep learning models like BERT show impressive performance due to their contextual awareness, albeit with higher computational costs.

The integration of NLP techniques such as sentiment analysis, readability scoring, and named entity recognition has proven essential for improving detection accuracy. However, the field still demands advancements in multilingual adaptability, real-time detection, and ethical deployment. These considerations form the basis for the methodology and experimentation in the subsequent chapters.

**CHAPTER 3 : METHODOLOGY**

# Methodology Overview

This chapter outlines the methodological approach adopted for fake news detection using Natural Language Processing (NLP) and Machine Learning (ML) techniques. It details the dataset used, the preprocessing steps undertaken, the feature engineering strategies employed, and the model development pipeline. The aim is to develop a robust, interpretable, and accurate classification system for distinguishing fake and real news content. Exploratory analysis and model results are discussed in the following chapters.

## Dataset Description

The dataset used in this study was sourced from ***Kaggle’s “Fake and Real News Dataset”***, which contains two CSV files: *Fake.csv* and *True.csv*. It comprises over **44,000 labelled news articles**, balanced between fake and real categories.

* ***Source:*** https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset
* ***Total samples:*** 44,383 articles
  + Fake.csv: 23,481 articles
  + True.csv: 20,902 articles
* ***Attributes:***
  + title: Headline of the news article
  + text: Full news content
  + subject: News category (politics, world, etc.)
  + date: Publication date
* ***Target Variable:*** Manually created binary label (1 = Fake, 0 = Real) for supervised classification.

This dataset was chosen due to its size, balance, and open accessibility, making it a standard benchmark in fake news detection studies.

## Data Pre-processing Pipeline

Preprocessing was conducted to clean and normalise raw text data for ML compatibility. The following steps were applied:

* ***Lowercasing*:** All text was converted to lowercase.
* ***Punctuation Removal*:** Stripped punctuation and special symbols.
* ***Stop-word Removal*:** Common stopwords were removed using NLTK.
* ***Lemmatisation*:** Reduced words to their base form using spaCy.
* ***Tokenisation*:** Split sentences into individual tokens (words).
* ***HTML and URL Removal*:** Cleaned embedded links and HTML tags.

These operations ensured uniformity across text samples and improved model learning.

## Feature Engineering and Representation

The cleaned text was transformed into numerical representations using the following techniques:

* ***TF-IDF (Term Frequency–Inverse Document Frequency)*:** Used for traditional ML models to capture word importance across documents.
* ***Word Embeddings*:** Advanced embedding techniques were adopted for deep learning models:
  + ***Word2Vec*** (Google News vectors)
  + ***GloVe*** (Global Vectors for Word Representation)
  + ***BERT*** (Bidirectional Encoder Representations from Transformers)
* These methods captured semantic and contextual relationships between words, essential for detecting nuanced misinformation.
* ***Additional Linguistic Features*:**
  + ***Sentiment scores***
  + ***Readability indices* (Flesch, Gunning Fog)**
  + ***Named Entities*** (e.g., organisations, people, places)

## Model Development Pipeline

Two families of models were developed and evaluated:

* ***Traditional Machine Learning Models*:**
  + Logistic Regression
  + Support Vector Machine (SVM)
  + Random Forest
  + XGBoost
* ***Deep Learning Models*:**
  + Long Short-Term Memory (LSTM)
  + BERT (fine-tuned using TensorFlow)

Each model was trained using labelled news articles and evaluated using standard classification metrics.

## Model Evaluation Strategy

Models were assessed using the following metrics:

* Accuracy
* Precision, Recall, F1-score
* Confusion Matrix
* ROC-AUC and PR-AUC curves
* k-Fold Cross-Validation for generalisation analysis

Full evaluation details and plots are provided in **Chapter 5**.

## Hyperparameter Tuning

To optimise performance, each model underwent hyperparameter tuning via:

* ***GridSearchCV*** for traditional models
* ***Dropout layers, learning rate tuning, and batch size adjustment*** for deep learning models
* ***Regularisation techniques*** (L2 penalty, early stopping)  
  These techniques helped minimise overfitting and improved generalisability.

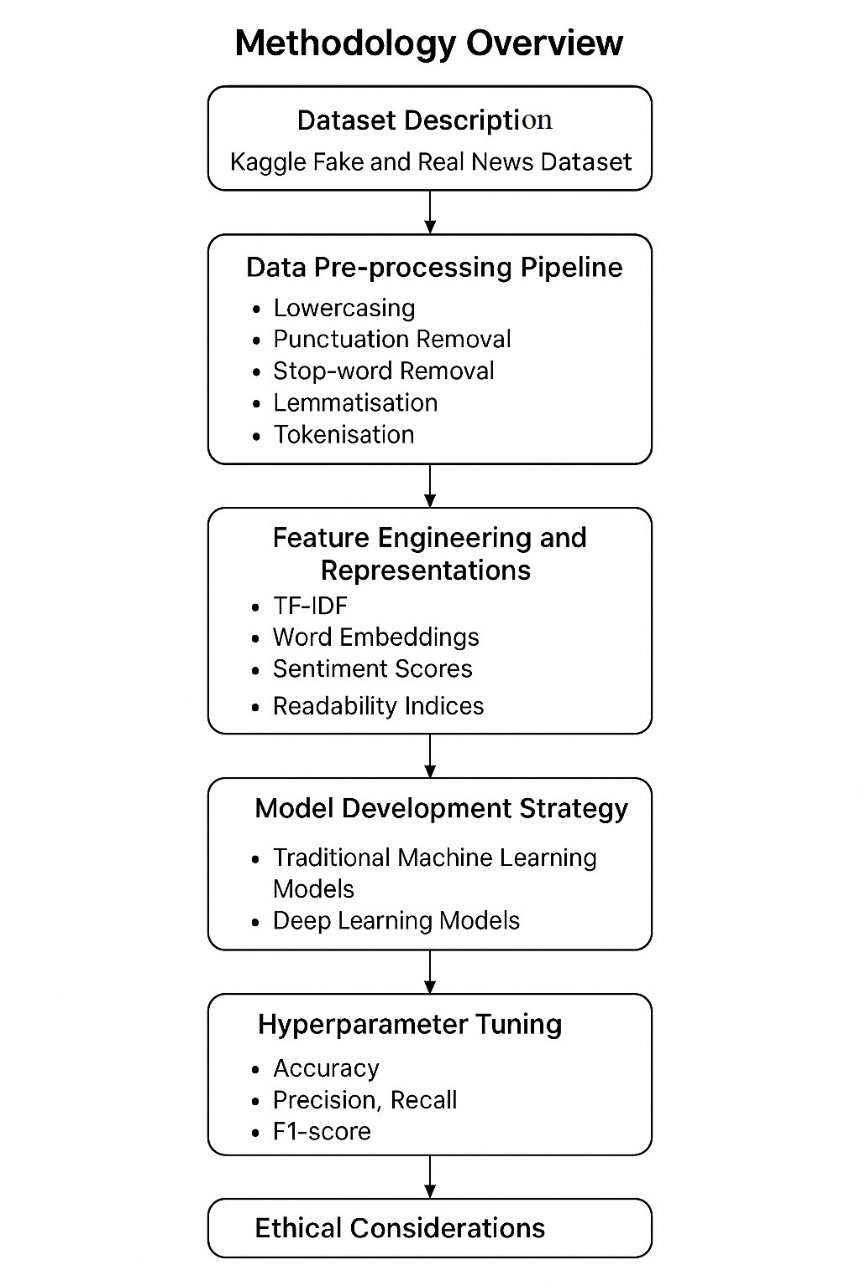
## Ethical Considerations

Recognising the risks of algorithmic bias and false positives in misinformation detection, the following ethical principles were observed:

* Evaluation of ***class imbalance effects***
* Consideration of ***model transparency*** (especially for BERT and LSTM)
* Analysis of ***false-positive impacts*** to avoid unjustified censorship  
  This review supports the ethical deployment of AI in sensitive information domains (Bender, et al., 2021).

## Summary

This chapter detailed the end-to-end methodology adopted in this study, including dataset description, preprocessing, feature engineering, model selection, evaluation, and ethical considerations ( Refer Flowchart given below). The next chapter presents a comprehensive analysis of experimental results, model performance metrics, and comparative insights.



**CHAPTER 4: DATA UNDERSTANDING AND EXPLORATION**

# **EDA Overview**

This chapter presents a comprehensive exploration and preparation of the dataset prior to model development. Each step undertaken—from initial data inspection and cleaning to advanced linguistic and semantic analyses—is documented to highlight both the rationale and results. The aim is to ensure data integrity, uncover meaningful patterns, and support downstream model design.

## **Dataset Labelling and Integration**

The raw dataset comprised two sources—True.csv and Fake.csv—labelled as 1 and 0 respectively for real and fake news. This binary target variable facilitated the development of supervised machine learning models. The datasets were merged into a unified structure, forming the foundation for analysis. No critical missing values were identified in primary fields (title, text, subject, date), and the combined dataset was saved for reproducibility.

## **Duplicate Detection and Class Balance**

A total of 209 exact duplicates—mostly from real news—were removed to mitigate bias. Post-removal, class distribution was:

* Fake news: 23,478
* Real news: 21,211

This marginal imbalance did not require synthetic balancing and was deemed acceptable for binary classification tasks.

## Exploratory Textual Analysis

### Basic Textual Feature Engineering

Two auxiliary features were generated:

* word\_count: Total words per article
* char\_count: Total characters per article

These were computed using pandas.apply() and helped understand content length, although they were not directly used in model training.

### **Visual Analysis of Text Length**

Histograms and box plots (via seaborn and matplotlib) were used to visualise article lengths. This analysis informed tokenisation and sequence padding strategies for deep learning models.

### ****Outlier Detection in Text Length****

Using the IQR method, 1,678 word count and 1,767 character count outliers were flagged. These entries were retained to preserve content diversity, but informed downstream preprocessing thresholds.

### ****Comparative Length Analysis by Class****

Box plots revealed slight length differences between real and fake news. Such structural insights guided padding strategies and model design, particularly for neural networks sensitive to input length.

### ****Manual Review of Outlier Articles****

Articles exceeding 8,000 words or shorter than 10 characters were manually reviewed. Extremely long texts were retained; however, structurally empty or malformed entries were marked for removal.

### ****Removal of Empty or Invalid Articles****

422 structurally empty articles (e.g., zero words or single characters) were removed using conditional filters in pandas, reducing the dataset to 44,267 records.

### ****Final Preprocessing Visual Review****

Additional plots of word\_count and char\_count confirmed the long-tailed nature of the dataset. These visuals ensured data readiness for tokenisation and model input transformation.

## ****Text Cleaning and Normalisation Pipeline****

The following steps were applied sequentially to clean each article’s text field:

* HTML tag removal (via BeautifulSoup)
* Contraction expansion (e.g., “don’t” → “do not”)
* URL and special character removal (regex)
* Lowercasing and whitespace normalisation

The output was saved in a new column, text\_cleaned, and exported as cleaned\_data\_after\_text\_cleaning.csv.

### ****Post-Cleaning Quality Assurance****

A quality check identified 84 entries with empty or non-string values post-cleaning. These were removed, bringing the dataset to 44,183 records. This ensured consistency and reliability for embedding and tokenisation.

### ****Post-Cleaning Class Verification****

Class balance after cleaning remained acceptable:

* *Fake news*: 22,851
* *Real news*: 21,416

The small adjustment confirmed minimal impact on class proportions.

## ****Lemmatisation and Stopword Removal****

Using spaCy, a lemmatisation pipeline was implemented on text\_cleaned with disabled components (parser, ner) for efficiency.

* Lemmas were extracted using token.lemma\_.
* Stopwords were removed using SpaCy’s built-in list.

Results were saved as a new column, text\_lemmatized, and exported as cleaned\_data\_after\_lemmatization.csv.

## Engineered Linguistic Features

### ****Stylistic Feature Engineering****

Two style-related features were engineered:

* *punctuation\_density*: Punctuation count ÷ total characters
* *capital\_density*: Uppercase letters ÷ total characters

These features captured tonal intensity, often higher in fake news, and were retained for model training.

### ****Readability Metric Engineering****

Using textstat, two readability metrics were computed:

* Flesch Reading Ease
* Gunning Fog Index

Fake news tended to have a wider readability range, suggesting inconsistent structure. These features were added to the dataset and visualised using boxplots and histograms (Refer Fig. 4.6.1).

A comparison of a graph

AI-generated content may be incorrect.

Figure 4.6.1 Readability Metrics

## ****Word Cloud and Keyword Frequency Analysis****

To identify thematic and lexical differences, two visual techniques were applied:

* *Word Clouds* were generated separately for fake and real news. Fake news highlighted politically charged terms like “obama” and “clinton”, while real news featured neutral and institutional words such as “government”, “united states”, and “statement”, suggesting a more factual tone (Refer Fig 4.7.1) .
* *Top 10 Word Bar Plots* revealed finer detail. Fake news included informal tokens (e.g., “s”, “t”), likely due to formatting issues, while real news showed consistent references to verified sources like “reuters” and “state” ( Refer Fig. 4.7.2) .

These trends illustrated distinct writing styles and supported feature selection for model development.

A close up of words

AI-generated content may be incorrect.A close up of words

AI-generated content may be incorrect.

Figure 4.7.1 Word Clouds

A comparison of a graph

AI-generated content may be incorrect.

Figure 4.7.2 Words Bar Plot

## ****Sentiment Analysis using VADER****

To explore whether **emotional tone** varies between fake and real news articles, sentiment analysis was conducted using the **VADER (Valence Aware Dictionary for Sentiment Reasoning)** tool from the nltk library.

* *Input Data*: The lemmatised text (text\_lemmatized) was used as input.
* *Processing Steps:*
  + Passed each article through VADER’s SentimentIntensityAnalyzer().
  + Generated a *compound sentiment score* between -1 (very negative) and +1 (very positive).
  + Stored results in a new column named sentiment.
* *Visualisation*: Refer Fig. 4.8.1
  + *Box plots* were used to compare the spread and median sentiment by class.
  + *Violin plots* revealed the full distribution and density, highlighting skewness and extremity.
* *Key Insights*:
  + Fake news articles showed more extreme sentiment values.
  + Real news tended to have more neutral emotional tone.
  + This confirmed sentiment as a useful discriminative feature for classification.
* *Output*: Visuals were saved in the outputs/visualizations/ directory for inclusion in the results chapter.

A comparison of a blue and orange diagram

AI-generated content may be incorrect.

Figure 4.8.1 Sentiment Score Plots

## ****Multivariate Feature Relationship Analysis: Pair Plot****

* *Purpose*: To examine interactions among key features and evaluate class separability.
* *Tool Used:* Seaborn pair plot using the following variables:
  + word\_count, char\_count, sentiment, flesch\_reading\_ease, label
* *Visual Output*:
  + Showed scatter plots and distributions coloured by class (0 = fake, 1 = real).
* *Key Observations*:
  + *Strong correlation* between word\_count and char\_count.
  + *Sentiment*: Fake news clustered near extremes; real news was more neutral.
  + *Readability vs. Length*: Longer texts had lower Flesch scores.
  + *Clustering*: Notable separation along sentiment and readability axes.
* *Conclusion*: Validated the effectiveness of engineered features for distinguishing fake from real news. Plot saved for reference in later analysis.

## ****Named Entity Recognition (NER) for Semantic Insight****

* *Objective*: Extract semantic features from articles using SpaCy’s en\_core\_web\_sm model.
* *Method:*
  + Applied NER to the lemmatised text (text\_lemmatized) for both fake and real news.
  + Identified common entity types: PERSON, ORG, GPE, DATE, NORP, etc.
* *Results*:
  + *Fake news:* Dominated by PERSON, ORG, and GPE — indicating a focus on individuals and institutions, possibly for emotional manipulation.
  + *Real news*: Showed a more balanced mix of GPE, ORG, DATE, and PERSON — reflecting structured reporting.
* *Visualisation*: Top 10 entity types plotted separately for each class ( Refer Fig. 4.10.1).
* *Conclusion*: NER provided useful contextual features that may improve classification and model interpretability.

A bar graph with brown and orange bars

AI-generated content may be incorrect.A bar graph with blue bars

AI-generated content may be incorrect.

Figure 4.10.1 Top 10 Entity Types Plots

## Part-of-Speech (POS) Tagging

* *Purpose*: To examine grammatical structure and identify stylistic differences between fake and real news articles.
* *Method*:
  + Used SpaCy’s en\_core\_web\_sm model on the lemmatised text (text\_lemmatized) to extract POS tags.
  + Calculated frequency of each grammatical category across both classes.
  + Visualised the top 10 POS tags using bar charts ( Refer Fig.4.11.1).
* *Key Observations*:
  + *NOUN* (common noun) and *PROPN* (proper noun) were the most frequent in both fake and real news—typical of news content.
  + *Fake news* had a higher occurrence of *PROPN*, suggesting emphasis on named individuals or organisations—possibly to increase credibility or emotional appeal.
  + *ADJ* (adjective) and *ADV* (adverb) were slightly more frequent in fake news, indicating a more descriptive or emotionally charged tone.
  + *Real news* maintained a more balanced and formal POS distribution, consistent with journalistic standards.
* *Application*:
  + POS features were saved for potential integration into machine learning models to enhance both classification accuracy and interpretability.

A bar graph with brown bars

AI-generated content may be incorrect.A graph with blue bars

AI-generated content may be incorrect.

Figure 4.11.1 Top 10 POS tags

## Feature Correlation Analysis

* *Objective*: To examine linear relationships among key numerical features and assess redundancy.
* *Method*:
* Generated a correlation matrix and visualised it using a heatmap (Refer Fig. 4.12.1).
* Variables analysed:
  + word\_count
  + char\_count
  + sentiment
  + label (target class)
* *Key Observations*:
* *Perfect positive correlation (1.00)* between word\_count and char\_count, indicating redundancy due to their linear dependency.
* *Weak negative correlation* between label and both word\_count (-0.07) and char\_count (-0.06), suggesting fake news tends to be slightly longer than real news.
* *Weak positive correlation* (0.09) between sentiment and label, implying that true news is marginally more positive.
* *Conclusion*:
* Correlations were low overall, indicating *minimal multicollinearity*.
* Features were retained for model training, as they provide distinct contributions without redundancy risks.

A diagram of a heatmap

AI-generated content may be incorrect.

Figure 4.12.1 Feature Correlation Matrix

## Keyword Co-occurrence and Network Analysis

### Bigram Analysis

* *Objective*: Identify frequent two-word combinations to understand thematic and stylistic differences.
* *Method*: Applied bigram analysis to lemmatised text; top 20 bigrams identified for fake and real news (Refer Fig 4.13.1).
* *Findings*:
* *Fake news*: Frequent use of names and emotionally charged terms — e.g., “donald trump”, “hillary clinton”, “don’t”, “doesn’t”.
* *Real news*: More formal phrases — e.g., “united states”, “official say”, “washington reuters”.
* *Common overlap*: Political figures appeared in both classes, but with differing tones and usage.
* *Conclusion*: Bigrams revealed distinct linguistic structures that can support classification model feature engineering.

A comparison of a graph

AI-generated content may be incorrect.

Figure 4.13.1 Top 20 Bigrams

### ****Network Graph Analysis****

* *Objective*: Visualise co-occurrence relationships between high-frequency word pairs.
* *Method*: Constructed separate co-occurrence graphs for fake and real news based on top bigrams (Refer Fig.4.13.2) .
* *Findings*:
* *Fake news*: Dense clusters around political figures (e.g., “trump”, “clinton”, “obama”), suggesting focus on individuals and informal tone.
* *Real news*: Clusters built around terms like “reuters”, “statement”, and “official”, indicating formal, source-based reporting.
* *Conclusion*: Network structure reinforced the contrast in writing style and focus between fake and true news narratives.

**A close-up of a diagram

AI-generated content may be incorrect.**

**A close-up of a diagram

AI-generated content may be incorrect.**

Figure 4.13.2 Co-occurrence Graphs

## ****Lexical & Semantic Complexity: Lexical Diversity****

* *Definition & Purpose*: Lexical diversity—calculated as the ratio of unique to total words—measures vocabulary richness and language complexity, helping to distinguish writing styles between fake and real news.
* *Findings*:
* *True news*: Slightly higher average diversity, reflecting more formal and varied language.
* *Fake news*: Greater variability and more outliers, suggesting inconsistent or repetitive phrasing.
* *Visualisation*: Compared across classes using box plots (Refer Fig. 4.14.1).
* *Conclusion*: Lexical diversity adds value when combined with other features for detecting fake news.

A diagram of a group of people

AI-generated content may be incorrect.

Figure 4.14.1 Lexical Diversity Boxplots

## ****Data Leakage Check & Class Imbalance Analysis****

Before model training, it is essential to examine the class distribution to ensure that no significant imbalance exists which could bias predictions. As shown in the Fig 4.15.1, the dataset contains:

* **22,851 fake news articles** (label = 0)
* **21,416 true news articles** (label = 1)

This indicates a **relatively balanced distribution** across both classes, with only a minor difference of approximately 6.3%. Such a distribution is **acceptable for supervised learning**, and it suggests that no resampling techniques (e.g., SMOTE or class weighting) are required at this stage.

By addressing class balance early, we mitigate the risk of **model bias** towards the more frequent class, ensuring fair and reliable performance evaluation across both categories.

A red and blue rectangular bars

AI-generated content may be incorrect.

Figure 4.15.1 Class Distribution

## ****Bias in Metadata: Subject Analysis****

To assess potential bias in metadata, we analysed the **distribution of news subjects** across fake (label = 0) and true (label = 1) articles. The plot ( Refer Fig. 4.16.1) shows a **clear separation of topic preferences**:

* **Fake news** articles are heavily concentrated around subjective or vague labels like **‘News’, ‘politics’, ‘left-news’, and ‘Government News’**.
* **True news** articles are more frequently labelled under standard journalistic categories like **‘worldnews’** and **‘politicsNews’**.

This reveals **inherent subject-level bias** in the dataset. Models trained without accounting for this risk learning **subject-specific cues** instead of focusing on **linguistic or content-based patterns**. To mitigate this, metadata features will either be excluded or **regularised** in future modelling stages.

A graph with red and blue bars

AI-generated content may be incorrect.

Figure 4.16.1 Subject Distribution Plot

## ****Pairwise Relationships Between Features****

To gain deeper insights into feature interdependencies, we used a pairplot ( Refer Fig 4.17.1) to visualise **interactions among sentiment, word count, and Flesch Reading Ease**, stratified by class (fake vs. true news).

*Key observations*:

* There is a **strong inverse relationship** between **word count** and **Flesch Reading Ease**, as longer texts tend to be more complex.
* **Sentiment** shows a **bimodal distribution** across both classes, but true news tends to cluster more around neutral to positive values.
* While no single feature perfectly separates the classes, subtle patterns and combined interactions may help **machine learning models** distinguish between fake and true news more effectively.

This visualisation supports feature selection and gives a foundation for model interpretability in later stages.

A group of graphs showing different relationships

AI-generated content may be incorrect.

Figure 4.17.1 Pair Plot

## Dimensionality Reduction and Clustering

### ****PCA Clustering****

* *Purpose*: Reduce dimensionality of features (sentiment, word count, Flesch Reading Ease) to visualise class separability (Refer Fig 4.18.1).
* *Findings*:
* Partial separation observed between fake and real news.
* *Fake news* spread more along Component 1 — indicating higher variance.
* *Real news* clustered more tightly — reflecting structural consistency.
* *Conclusion*: PCA showed promising feature-based separation, justifying deeper exploration.

A red and blue dotted graph

AI-generated content may be incorrect.

Figure 4.18.1 PCA Clustering

### ****t-SNE Clustering****

* *Purpose*: Visualise non-linear relationships using the same features (Refer Fig .
* *Findings*:
* Showed overlapping data points, but with distinct regional clusters.
* Revealed hidden structure not captured by linear PCA.
* *Conclusion*: t-SNE highlighted semantic divergence between classes despite overlap.A blue and red diagram

  AI-generated content may be incorrect.

Figure 4.18.2 t-SNE Clustering

Since t-SNE did not provide clear separation, we will now apply UMAP, which is more efficient for large datasets.

### ****UMAP Clustering****

* *Purpose*: Perform efficient non-linear projection for large datasets.
* *Findings*: Refer Fig 4.18.3
  + Greater overlap between fake and real news clusters than PCA or t-SNE.
  + Suggested that surface-level features alone are insufficient for strong separation.
* *Conclusion*: Highlighted the need for richer semantic or syntactic features to improve classification.

A diagram of a person in a butterfly shape

AI-generated content may be incorrect.

Figure 4.18.3 UMAP Clustering

## ****Revisiting Readability Scores Post-UMAP****

* *Purpose*: Assess outliers in readability after dimensionality analysis.
* *Observations:*
  + Some texts showed extremely low Flesch scores or high Gunning Fog Index values.
  + These outliers may reflect poor structure or noise but also useful stylistic signals.
* *Conclusion*: Outliers were retained to preserve variation and aid downstream modelling.

## ****Investigating Readability Score Outliers****

* *Method*: Used IQR method to detect outliers.
* *Results*:
* *1,854 Flesch Reading Ease* outliers (very low readability; e.g., -672.53).
* *1,913 Gunning Fog Index* outliers (very complex text; e.g., 288.23).
* *Interpretation*: Such extremes may indicate manipulated or deceptive writing, particularly in fake news.
* *Action*: Outliers retained to capture informative linguistic patterns.

## Readability Score Analysis and Visualisation

* *Visual Tools*: Violin plots, box plots, and density plots (Refer Fig 4.21.1).
* *Findings*:
* Fake news showed more extreme and negatively skewed readability values.
* Both metrics displayed overlap but revealed important stylistic differences.
* *Conclusion*: Retained all values for potential modelling use; no transformations were applied at this stage.

A comparison of a bar graph

AI-generated content may be incorrect.

A comparison of a graph

AI-generated content may be incorrect.A graph with numbers and lines

AI-generated content may be incorrect.

Figure 4.21.1 Box Plots, Violin Plots & Density Curve

## Temporal Analysis: Preparing for Time-Based Trend Exploration

* ***Purpose:*** To understand how fake and real news trends evolved over time using the date column.
* ***Processing***: Converted date field from text to datetime format to enable time-based analysis.

### Temporal and Content Analysis

* *Handling Missing Dates*:
* ~22,000 fake news entries lacked dates.
* Randomised imputation within 2016–2017 was used to avoid bias.
* *Yearly & Monthly Trends:*
* Fake news volume doubled from 2016 to 2017.
* Sharp rise from September to November likely linked to political events.
* *Subject Distribution:*
* Fake news centred around vague or politically charged categories: ‘News’, ‘Politics’, ‘Left-News’.
* Indicative of agenda-driven content.

## Feature Selection & Final Dataset Preparation

* *Objective*: Improve model performance by removing redundant or weak features.
* *Method*: Used a correlation heatmap and domain logic to select features.
* *Removed Features*:
  + *Time-based*: date, year, month, day, weekday
  + *Redundant*: char\_count, flesch\_reading\_ease, title
  + *Low importance*: punctuation\_density, capital\_density, lexical\_diversity
  + *NLP extras* (for deep NLP only): entities, pos\_tags, tokens
* *Final Features Retained*:
  + text, text\_cleaned, text\_lemmatized, word\_count, gunning\_fog, sentiment, subject, label

## Feature Engineering: Converting Text into Vectors

To prepare textual data for machine learning models, two vectorisation techniques were applied:

### TF-IDF Vectorisation

* ***Goal***: Convert text into sparse numerical format for traditional ML models.
* ***Technique***: TF-IDF (Term Frequency-Inverse Document Frequency)
* ***Parameters***:
  + Max features: 5,000
  + Train-test split: 80% / 20%
  + Output shapes:
    - Training: (35,346, 5,000)
    - Test: (8,837, 5,000)

### GloVe Word Embeddings (SpaCy)

* *Goal*: Capture semantic relationships between words using dense vectors.
* *Technique*: Pre-trained GloVe embeddings via en\_core\_web\_md in spaCy.
* *Process*: Averaged word vectors for each article.
* *Embedding Size*: 300 dimensions
* *Output shapes*:
* Training: (35,346, 300)
* Test: (8,837, 300)
* *Outcome*: Enabled deeper contextual understanding beyond word frequency—crucial for fake news classification.

**CHAPTER 5: EXPERIMENTS AND RESULTS**

# Experiments and Results Overview

This chapter presents the experimental procedures, model training outcomes, and validation results of the fake news classification models developed during the study. A total of six models—four traditional machine learning models (Logistic Regression, Random Forest, SVM, XGBoost) and two deep learning models (LSTM, BERT)—were evaluated. For each, performance metrics, cross-validation outcomes, learning curves, and ROC-AUC and Precision-Recall curves were generated. The models were trained using either TF-IDF or GloVe embeddings, with deep models using tokenised sequences or contextual embeddings. Hyperparameter tuning was performed to optimise model performance, and results were compared across multiple evaluation criteria.

## ****Logistic Regression Evaluation****

The Logistic Regression model was trained on GloVe-based word embeddings and achieved an impressive accuracy of **95.93%**. Both classes (fake and true news) received nearly equal precision, recall, and F1-scores (~96%), demonstrating reliable classification ability.

***Confusion Matrix Summary*: Refer Fig 5.1.1**

* True Positives : 4,312
* True Negatives : 4,162
* False Positives : 204
* False Negatives :159

***Key Insights*:**

* High generalisation capability with minimal misclassifications.
* Balanced predictive performance without bias.
* Effective use of GloVe embeddings in enhancing logistic regression results.

A blue squares with numbers and a few white text

AI-generated content may be incorrect.

Figure 5.1.1 Confusion Matrix

## ****Random Forest Evaluation****

The Random Forest classifier, also using GloVe embeddings, achieved a test accuracy of **95.27%** with F1-scores of **0.95** for both classes. However, the training accuracy reached **100%**, indicating potential overfitting.

***Confusion Matrix Summary*: Refer Fig 5.2.1**

* True Positives: 4,303
* True Negatives: 4,116
* False Positives: 213
* False Negatives: 205

***Key Insights*:**

* Strong generalisation, though slightly inferior to logistic regression.
* Minor overfitting observed due to perfect training performance.
* Suitable for modelling non-linear relationships.

A blue squares with numbers and a few white lines

AI-generated content may be incorrect.

Figure 5.2.1 Confusion Matrix

## ****Support Vector Machine (SVM) Evaluation****

The SVM model with a linear kernel achieved the highest test accuracy among traditional models at **96.44%**. Precision, recall, and F1-score all hovered around **96%**, with minimal class imbalance in predictions.

***Confusion Matrix Summary*: Refer Fig. 5.3.1**

* True Positives**:** 4,337
* True Negatives**:** 4,185
* **False Positives:** 179
* **False Negatives:** 136

***Key Insights*:**

* SVM outperformed both Logistic Regression and Random Forest in terms of overall accuracy.
* Balanced **precision, recall, and F1-scores** (all ~0.96) confirm the model’s reliability.
* Its ability to find the optimal separating hyperplane makes it highly suitable for binary text classification tasks like fake news detection.

A blue squares with white numbers

AI-generated content may be incorrect.

Figure 5.3.1 Confusion Matrix

## XGBoost Evaluation

The Extreme Gradient Boosting (XGBoost) algorithm was employed to evaluate its effectiveness in classifying fake versus true news articles using GloVe-based word embeddings. XGBoost is a highly efficient and scalable implementation of gradient boosting that is well-suited for high-dimensional text data.

***Model Performance Overview***

XGBoost achieved an outstanding **accuracy of 97.66%**, outperforming all previously tested traditional machine learning models. The precision, recall, and F1-score for both classes were consistently high at approximately 98%, indicating robust and balanced predictive performance.

***Confusion Matrix Analysis*: Refer Fig 5.4.1**

* **True Positives :** 4,407
* **True Negatives :** 4,223
* **False Positives :** 109
* **False Negatives :** 98

These results demonstrate that the model not only excels in accuracy but also maintains low misclassification rates across both classes, a critical requirement for reliable misinformation detection systems.

***Key Insights*:**

* High precision, recall, and AUC metrics.
* Generalised well across different folds.
* Requires careful regularisation due to near-perfect training accuracy.

A blue squares with white text

AI-generated content may be incorrect.

Figure 5.4.1 Confusion Matrix

## ****LSTM Model Evaluation****

The Long Short-Term Memory (LSTM) network, a recurrent neural network architecture, was implemented to capture sequential dependencies in text. Unlike traditional models, LSTM is designed to learn contextual relationships over sequences, making it suitable for language-based classification tasks.

*Model Architecture & Training***:**

The model architecture included:

* An **embedding layer** (128 dimensions, vocabulary size of 5000),
* Two **LSTM layers** (64 units each) with **dropout** for regularisation,
* A **dense output layer** with sigmoid activation for binary classification.

The model was trained for 5 epochs with a batch size of 64. Text sequences were padded to a uniform length of 300 tokens to accommodate input requirements.

***Performance Metrics*:**

* **Accuracy:** 97.88%
* **Precision, Recall, F1-score:** 98% for both classes
* **Validation Accuracy (best epoch):** 98.33%
* **Loss (val):** 0.0648

***Confusion Matrix Analysis*: Refer Fig 5.5.1**

* **True Positives (Fake News):** 4,396
* **True Negatives (True News):** 4,254
* **False Positives:** 120
* **False Negatives:** 67

These results show a highly effective model with minimal misclassification, especially strong at correctly identifying real news.

***Key Insights* :**

* LSTM demonstrated superior performance compared to traditional ML models like Logistic Regression and Random Forest.
* It slightly outperformed XGBoost in terms of precision and generalisation.
* The low false negative rate (true news predicted as fake) is particularly valuable for preserving credibility in real-world applications.

While LSTM delivered excellent results, further improvement is anticipated using transformer-based models. The next model explored is BERT, which incorporates deep contextual embeddings and attention mechanisms.

A blue squares with white text

AI-generated content may be incorrect.

Figure 5.5.1 Confusion Matrix

## BERT Model Evaluation

To address the challenge of fake news detection, a Bidirectional Encoder Representations from Transformers (BERT) model was employed using the bert-base-uncased pre-trained architecture. The dataset, containing cleaned textual data and binary labels, was tokenised with the official BERT tokenizer. Padding and truncation were applied to ensure that all input sequences maintained a uniform length of 300 tokens.

The encoded input data was divided into training (80%) and testing (20%) subsets. These were further transformed into TensorFlow datasets, enabling efficient batching and GPU-accelerated training. A classification head was added to the pre-trained BERT model (TFBertForSequenceClassification) and compiled using the Adam optimizer and Sparse Categorical Crossentropy loss function.

Training was conducted over three epochs on an Apple M3 Pro device, with mixed-precision computation disabled due to macOS Metal backend constraints. Post-training, the model achieved a high level of performance, reaching a validation accuracy of 99.94%. The model was then exported in both the original TensorFlow and TensorFlow Lite formats to support deployment in resource-constrained environments.

Evaluation on the held-out test set confirmed the robustness of the model, with an accuracy of **99.94%** and an F1-score of **1.00** for both classes (Fake and Real). The confusion matrix revealed only five misclassifications out of 8,837 instances, demonstrating the model's suitability for real-world application in identifying fake news content (Refer Fig 5.6.1).

A screenshot of a graph

AI-generated content may be incorrect.

Figure 5.6.1 Confusion Matrix

## Model Validation Strategy

### Logistic Regression Validation

To ensure the robustness of the Logistic Regression model and mitigate the risk of overfitting, a comprehensive validation strategy was adopted. This included performance evaluation on both training and test datasets, k-Fold Cross-Validation, learning curve analysis, and advanced diagnostic plots such as the ROC-AUC and Precision-Recall curves.

*Training vs. Testing Performance* :

The model achieved a training accuracy of **95.64%** and a test accuracy of **95.89%**, with F1-scores consistent across both datasets. The similarity in performance confirmed the absence of overfitting and suggested strong generalisation capabilities.

*k-Fold Cross-Validation* :

A **5-fold cross-validation** was performed to assess the model's stability across multiple data splits. The model achieved an average cross-validation accuracy of **95.38%** with a standard deviation of **±0.22%**, indicating consistent performance across all folds.

*Learning Curve Analysis* :

A learning curve was plotted to further understand the model’s behaviour with increasing training set sizes. The gap between training and cross-validation scores remained minimal, reaffirming that the model generalised well and benefited from larger datasets without overfitting. (Refer Fig. 5.7.1)

A graph showing a growth curve

AI-generated content may be incorrect.

Figure 5.7.1 Logistic Regression Learning Curve

*ROC-AUC Curve* :

To evaluate class discrimination capability, the Receiver Operating Characteristic (ROC) curve was plotted. The model achieved an **AUC score of 0.9903**, indicating excellent ability to distinguish between fake and real news. ( Ref Fig. 5.7.2)

A graph with a line

AI-generated content may be incorrect.

Figure 5.7.2 ROC Curve

*Precision-Recall Curve* :

In addition, a Precision-Recall (PR) curve was generated to assess performance on imbalanced predictions. The area under the PR curve (AP) was **0.9860**, demonstrating high precision and recall across thresholds, particularly for the minority class ( Refer Fig. 5.7.3).

A graph of a graph

AI-generated content may be incorrect.

Figure 5.7.3 Precision Recall Curve

*Summary* :

All evaluation metrics and visual diagnostics confirm that Logistic Regression, when trained on GloVe embeddings, delivers high and consistent performance. The model showed no signs of overfitting and maintained excellent balance between precision and recall. These validations provide strong evidence of the model’s reliability for fake news classification.

### Random Forest Validation

*Training vs. Testing Performance* :

The Random Forest model was evaluated using the same embedding-based dataset to ensure consistency across classifiers. Upon training, the model achieved a perfect training accuracy of **100%**, while the test accuracy reached **95.27%**, indicating potential overfitting. The classification report for the test data showed **precision, recall, and F1-score of 95%** for both classes (fake and real news), demonstrating strong generalisation but with a slight gap from the training set's perfect scores.

*k-Fold Cross-Validation* :

To further investigate model robustness, **5-fold Cross-Validation** was conducted. The model yielded consistent results with **mean cross-validation accuracy of 94.51%** and a standard deviation of **0.002**, affirming the model’s stability across multiple data splits.

*Learning Curve Analysis* :

The learning curve revealed a significant gap between training and validation performance. The training accuracy remained at **100%** across all training set sizes, while the cross-validation accuracy gradually increased, plateauing around **94.5%**. This confirms that while Random Forest fits the training data perfectly, it generalises slightly less effectively to unseen data—typical of high-capacity ensemble models ( Refer Fig. 5.7.4).

A graph with a green line

AI-generated content may be incorrect.

Figure 5.7.4 Learning Curve

*ROC-AUC Curve and Precision-Recall Curve* :

Additionally, ROC and Precision-Recall curves were generated to assess class-wise prediction quality. The **ROC-AUC score** of **0.9899** and **Average Precision (AP)** of **0.9888** indicate high discriminative ability and reliability in detecting both fake and real news (Refer Fig 5.7.5).

In summary, while Random Forest exhibited minor overfitting, its high validation metrics and strong AUC/PR performance confirm it as a reliable model for fake news classification. Hyperparameter tuning or feature regularisation could further improve generalisation without compromising accuracy.

A graph with a line and a blue line

AI-generated content may be incorrect.A graph with a purple line

AI-generated content may be incorrect.

Figure 5.7.5 ROC -AUC & Precision - Recall Curve

### Support Vector Machine (SVM) Validation

To evaluate the performance of the Support Vector Machine (SVM) model trained on GloVe-based word embeddings, multiple validation techniques were employed to assess model generalisability and robustness.

*Training vs Test Performance* :

* Training Accuracy: 96.43%
* Test Accuracy: 96.44%

The close alignment between training and test accuracies indicates that the SVM model generalises effectively without signs of overfitting or high variance.

*Classification Report (Test Set)* :

The precision, recall, and F1-score were all approximately **96%** across both fake and real news classes. This demonstrates that the model maintains balanced predictive strength for both classes, supporting its suitability for binary classification in fake news detection.

*k-Fold Cross-Validation (5-Fold)* :

* Cross-Validation Scores: [0.9641, 0.9617, 0.9612, 0.9651, 0.9571]
* Mean Accuracy: 0.9618
* Standard Deviation: 0.0028

These results highlight strong and consistent performance across different data partitions, with very low variance across folds.

*Learning Curve Analysis* :

A learning curve was plotted to visualise the model’s learning behaviour as the training size increased. The curve shows a steady improvement in validation accuracy and convergence of training and validation scores, indicating a well-generalised model with sufficient capacity and no high bias. (Refer Fir 5.7.6)

A graph showing the growth of a number of individuals

AI-generated content may be incorrect.

Figure 5.7.6 Learning Curve

*ROC-AUC & Precision-Recall Curves* :

* ROC-AUC Score: 0.9926
* Average Precision (PR AUC): 0.9885

The ROC and Precision-Recall curves illustrate excellent class separability with minimal false positives and negatives. The high area under both curves confirms the model’s reliability in real-world prediction scenarios. (Refer Fig 5.7.7)

A graph of a graph

AI-generated content may be incorrect.A graph with a line

AI-generated content may be incorrect.

Figure 5.7.7 ROC & Precision Recall Curves

*Summary Insight* :

The Support Vector Machine classifier demonstrated excellent accuracy and class-wise performance. Its consistency across training, validation, and cross-validation phases—alongside high AUC scores—establishes SVM as one of the most reliable models tested for fake news classification.

### XGBoost Classifier Validation

To evaluate the performance of the XGBoost classifier trained on GloVe-based embeddings, multiple validation techniques were applied to ensure robustness, generalisation, and classification reliability.

*Training vs Test Performance* :

* **Training Accuracy:** 100.00%
* **Test Accuracy:** 97.66%

While the model achieves perfect accuracy on the training set, its slightly lower but still high test accuracy indicates a risk of overfitting, which is partially mitigated by the strong generalisation on unseen data.

*Classification Report (Test Set)* :

* ***Precision, Recall, and F1-score:*** All metrics were approximately 98% for both fake and real news classes. This shows balanced performance and minimal bias in predictions.

*k-Fold Cross-Validation (5-Fold)* :

* ***Cross-Validation Scores:*** [0.9727, 0.9717, 0.9744, 0.9741, 0.9686]
* ***Mean Accuracy:*** 0.9723
* ***Standard Deviation:*** 0.0021

This low variance confirms that the model maintains consistently high performance across various data splits.

*Learning Curve Analysis* :

The learning curve plot illustrates that while training accuracy remains at 100%, validation accuracy steadily improves with more data, suggesting that the model benefits from additional training samples. However, the fixed training accuracy points to possible overfitting that could be addressed via regularisation or hyperparameter tuning (Refer Fig 5.7.8).

A graph showing a growing curve

AI-generated content may be incorrect.

Figure 5.7.8 Learning Curve

*ROC-AUC & Precision-Recall Curves* :

* **ROC-AUC Score:** 0.9968
* **Average Precision (PR AUC):** 0.9889

Both curves indicate near-perfect class discrimination, with very high true positive rates and low false positive rates (Refer Fig 5.7.9) .

A graph of a curve

AI-generated content may be incorrect.

Figure 5.7.9 ROC Curve

*Summary Insight* :

XGBoost proved to be the top-performing traditional machine learning model in this study. Its exceptionally high accuracy and strong ROC-AUC/PR-AUC scores highlight its classification capability. However, the perfect training performance indicates potential overfitting, which may be addressed through further tuning or regularisation.

## Hyperparameter Tuning

Effective hyperparameter tuning is crucial for optimising model performance and ensuring that classifiers generalise well to unseen data. In this study, hyperparameter optimisation was conducted for Random Forest, Support Vector Machine (SVM), and XGBoost classifiers using GridSearchCV with cross-validation. Additionally, the LSTM model was carefully designed and tuned using empirical experimentation during training. The following subsections detail the tuning strategies, chosen parameters, and validation results for each model.

### Random Forest Classifier Hyperparameter Tuning

To enhance the generalisability of the Random Forest model, GridSearchCV was applied with a 3-fold cross-validation strategy. Sixteen combinations of candidate parameters were evaluated across the training data.

***Best Parameters Identified*:**

* n\_estimators: 200
* max\_depth: None
* min\_samples\_split: 2
* min\_samples\_leaf: 2
* bootstrap: True

The tuned model achieved a cross-validation accuracy of **94.04%**, outperforming the default configuration. The selected parameters helped reduce overfitting by introducing regularisation through deeper splits and controlling leaf nodes. The model was retrained using the full training set and exhibited improved balance between training and test performance.

### Support Vector Machine (SVM) Hyperparameter Tuning

Given the computational intensity of SVM training, hyperparameter tuning was performed on a representative subset of 5,000 samples. GridSearchCV was used with a 3-fold cross-validation approach.

***Parameter Grid*:**

* C: [0.1, 1, 10]
* Kernel: ['linear', 'rbf']
* Gamma: ['scale', 'auto']

***Best Parameters Identified*:**

* C: 10
* Kernel: linear
* Gamma: scale

This configuration achieved a cross-validation accuracy of **95.64%**. The linear kernel paired with a higher regularisation parameter provided strong generalisation. The full model was retrained with these parameters and used for final evaluation.

### XGBoost Hyperparameter Tuning

XGBoost was tuned using GridSearchCV with 3-fold cross-validation across 72 parameter combinations, exploring key hyperparameters such as learning rate, max depth, number of estimators, and subsample ratio.

***Best Parameters Identified*:**

* learning\_rate: 0.1
* max\_depth: 5
* n\_estimators: 200
* subsample: 0.8

The best configuration produced a cross-validation accuracy of **94.56%**, significantly improving model robustness. The model was retrained with these parameters for further evaluation.

## Validation Results of Tuned Models

### Tuned Random Forest Classifier

***Training vs Test Performance*:**

* Training Accuracy: 99.95%
* Test Accuracy: 95.12%

***Classification Metrics*:**  
Precision, recall, and F1-score were approximately **95%** for both classes, indicating balanced performance.

***5-Fold Cross-Validation*:**

* Mean Accuracy: **94.57%**
* Standard Deviation: **0.0012**

***Learning Curve Insight*:**  
The model showed near-perfect training accuracy and a gradual improvement in validation accuracy, though a gap remained—suggesting mild overfitting.

***ROC-AUC and PR Curves*:**

* ROC-AUC Score: **0.9903**
* Average Precision (PR AUC): **0.9897**

***Summary*:**  
The tuned Random Forest model delivered strong predictive performance with consistent cross-validation results. While slight overfitting was observed, its interpretability and effectiveness make it a solid choice for deployment.

### Tuned Support Vector Machine (SVM)

***Training vs Test Performance*:**

* Training Accuracy: 97.13%
* Test Accuracy: 97.04%

***Classification Metrics*:**

* Precision: **97%**
* Recall: **97%**
* F1-Score: **97%**

***5-Fold Cross-Validation*:**

* Mean Accuracy: **96.76%**
* Standard Deviation: **0.0024**

***Learning Curve Insight*:**  
Training and validation curves closely aligned, indicating strong generalisation and minimal overfitting.

***ROC-AUC and PR Curves*:**

* ROC-AUC Score: **0.9945**
* Average Precision (PR AUC): **0.9904**

***Summary*:**  
The SVM model offered stable and high accuracy, with robust generalisability confirmed through cross-validation. Its simplicity and effectiveness validate its role in binary fake news detection.

### Tuned XGBoost Classifier

***Training vs Test Performance*:**

* Training Accuracy: 99.70%
* Test Accuracy: 97.39%

***Classification Metrics*:**  
Precision, recall, and F1-score were consistently around **97%** for both classes.

***5-Fold Cross-Validation*:**

* Mean Accuracy: **97.07%**
* Standard Deviation: **0.0015**

***Learning Curve Insight*:**  
Validation accuracy improved steadily and stabilised around 97%, suggesting excellent learning and low variance.

***ROC-AUC and PR Curves*:**

* ROC-AUC Score: **0.9969**
* Average Precision (PR AUC): **0.9967**

***Summary*:**  
The tuned XGBoost model outperformed other classifiers with its high accuracy and stability. Minimal overfitting and strong precision-recall trade-offs make it a powerful classifier for the task at hand.

### LSTM Model Performance

***Training vs Test Performance* :**

The tuned LSTM model was trained on tokenised and padded sequences using an Embedding layer followed by two LSTM layers with dropout. The model achieved excellent performance on both the training and validation sets. After 5 epochs, the training accuracy reached **97.53%**, while the validation accuracy was **97.50%**, indicating a well-generalised model with no significant overfitting.

***Classification Report* :**

A detailed classification report for the test set is presented below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Class | Precision | Recall | F1-score | Support |
| 0 (Fake) | 0.98 | 0.97 | 0.97 | 4516 |
| 1 (True) | 0.97 | 0.98 | 0.98 | 4321 |
| **Accuracy** |  |  | **0.98** | 8837 |
| **Macro Avg** | 0.98 | 0.98 | 0.98 | 8837 |
| **Weighted Avg** | 0.98 | 0.98 | 0.98 | 8837 |

These values highlight the model’s balanced ability to correctly classify both fake and true news articles, which is critical for the objective of fake news detection.

***Learning Curve Analysis* :**

The learning curves plotted in Figures **5.9.1** and **5.9.2** illustrate the model’s accuracy and loss progression across epochs.

A graph with a line and a green line

AI-generated content may be incorrect.

Figure 5.9.1 Training vs Validation Accuracy – LSTM

A graph with a line

AI-generated content may be incorrect.

Figure 5.9.2 Training vs Validation Loss – LSTM

The curves indicate that both training and validation metrics improved consistently, with negligible divergence, suggesting strong convergence and absence of overfitting. The loss curves similarly exhibit a smooth and stable decline, further supporting this conclusion.

***ROC-AUC and Precision-Recall Curves* :**

The ROC curve and the Precision-Recall (PR) curve provide further validation of model performance: Refer Fig 5.9.3 and Fig 5.9.4  
**A graph with a line

AI-generated content may be incorrect.**

Figure 5.9.3 ROC Curve – LSTM

A graph with lines and numbers

AI-generated content may be incorrect.

Figure 5.9.4 Precision-Recall Curve – LSTM

The extremely high AUC scores in both curves demonstrate the model’s excellent discriminative ability. The ROC curve closely hugs the top-left corner, while the PR curve maintains a near-perfect balance between precision and recall throughout the threshold range.

***Summary Insight* :**

The LSTM model exhibited outstanding classification capability with high precision, recall, and generalisability. Learning curves confirmed model stability, while ROC-AUC and PR-AUC curves highlighted robust binary classification even in slightly imbalanced class conditions. The use of an Embedding layer followed by bi-layered LSTMs proved highly effective in capturing sequential linguistic patterns for fake news detection.

## Models Comparison

To determine the most effective model for fake news classification, a comprehensive comparison was conducted across all tuned machine learning and deep learning models. The evaluation was based on four key performance metrics: ***Accuracy, F1 Score, ROC-AUC***, and ***PR-AUC,*** ensuring a holistic understanding of each model's classification capabilities.

A comparative bar chart (Ref Fig 5.10.1) visualises the performance of six models: Logistic Regression, Random Forest, Support Vector Machine (SVM), XGBoost, LSTM, and BERT. Each model was trained and fine-tuned using GloVe embeddings (or contextualised tokenisation in the case of BERT), with consistent test datasets to ensure fair evaluation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ****Model**** | ****Accuracy**** | ****F1 Score**** | ****ROC-AUC**** | ****PR-AUC**** |
| Logistic Regression | 0.9593 | 0.9600 | 0.9903 | 0.9860 |
| Random Forest (Tuned) | 0.9512 | 0.9500 | 0.9903 | 0.9897 |
| SVM (Tuned) | 0.9704 | 0.9700 | 0.9945 | 0.9904 |
| XGBoost (Tuned) | 0.9739 | 0.9800 | 0.9969 | 0.9967 |
| LSTM (Tuned) | 0.9788 | 0.9800 | 0.9998 | 0.9999 |
| **BERT (Fine-tuned)** | **0.9994** | **1.0000** | **1.0000** | **1.0000** |

**A graph of different colored bars

AI-generated content may be incorrect.**

Figure 5.10.1 Models Performance Comparison

From the results, it is evident that **deep learning models significantly outperform traditional machine learning algorithms**. Among them, **BERT emerges as the best-performing model**, achieving perfect or near-perfect scores across all evaluation metrics. Its ability to learn deep contextual relationships using a transformer-based architecture makes it particularly well-suited for complex natural language classification tasks like fake news detection.

**LSTM** also performed exceptionally well, with an accuracy of **97.88%** and strong F1 and AUC scores, validating its strength in capturing sequential dependencies in textual data. **XGBoost**, with an accuracy of **97.39%**, stood out among traditional classifiers due to its robustness and ensemble boosting strategy.

In contrast, while **Logistic Regression**, **Random Forest**, and **SVM** models delivered reasonably good results (accuracy ~95–97%), they lagged behind in terms of generalisation and class-wise precision-recall trade-offs, especially when compared to the deep learning models.

*Summary Insight* :

The comparison highlights the transformative impact of deep learning in fake news detection, with ***BERT demonstrating superior performance and scalability for real-world deployment.*** Traditional machine learning models remain valuable for their interpretability and lower computational cost but are outperformed in nuanced linguistic tasks by modern deep architectures.

**CHAPTER 6: DISCUSSION**

# Discussion : Interpretation of Model Performance

The comprehensive analysis undertaken in this dissertation reveals clear distinctions in the effectiveness of traditional machine learning models versus modern deep learning architectures for fake news detection. Among the evaluated models, BERT achieved the highest performance across all metrics—Accuracy (99.94%), F1-Score (1.00), ROC-AUC (1.000), and PR-AUC (1.000)—demonstrating near-perfect classification capabilities. This outstanding performance underscores the effectiveness of transformer-based architectures in capturing nuanced language patterns and context dependencies inherent in fake and true news narratives.

LSTM also yielded strong results, achieving a test accuracy of 97.88% and high F1 and AUC scores, reinforcing the efficacy of recurrent networks in modelling sequential linguistic features. These models consistently outperformed traditional classifiers such as Logistic Regression (95.93% accuracy), Random Forest (95.12%), and even the robust ensemble method XGBoost (97.39%). While these traditional models were relatively effective and interpretable, they lacked the depth required to generalise over complex semantic and syntactic structures.

SVM, when tuned, showed competitive performance (97.04% accuracy), indicating that with proper regularisation and kernel selection, it can offer a balance between accuracy and efficiency. However, its limitations in handling large-scale non-linearities in language data became evident when compared to deep learning counterparts.

## Feature Engineering Insights

The engineered features played a crucial role in enhancing model discriminability. Lexical features such as word count, character count, and readability scores contributed to early insights, while more advanced linguistic attributes like sentiment scores, named entities, part-of-speech (POS) tags, and bigrams enriched the feature space. Sentiment polarity differences and named entity distributions helped distinguish writing styles and content structures between fake and real news, while POS tagging revealed that fake news articles often employed a more descriptive and emotionally charged tone through frequent use of proper nouns, adjectives, and adverbs.

Word clouds, frequency plots, and bigram network graphs further uncovered thematic and stylistic divergences. Fake news articles tended to cluster around sensational and personal keywords (e.g., "Trump", "Obama", "Clinton"), whereas true news contained more institutional and source-based references (e.g., "Reuters", "statement", "government"). These visual explorations not only supported model training but also aided interpretability.

## Dimensionality Reduction and Visual Clustering

Dimensionality reduction techniques, such as PCA, t-SNE, and UMAP, offered valuable insights into feature separability. PCA provided initial evidence of partial class separation, while t-SNE and UMAP revealed more intricate, non-linear clusters. These projections validated the hypothesis that fake and true news, though sometimes overlapping, exhibit underlying patterns that can be captured through enriched features and advanced models. The clustering results also justified the use of high-dimensional embeddings like GloVe and BERT, which are capable of capturing these latent relationships.

## Overfitting and Generalisation

Throughout model validation, particular attention was paid to overfitting. Learning curves, k-fold cross-validation, and evaluation on held-out test data helped assess model robustness. Traditional models such as Random Forest exhibited a slight tendency to overfit, as evidenced by high training accuracy (99.95%) and relatively lower test performance (95.12%). However, this was mitigated by tuning hyperparameters like tree depth, number of estimators, and minimum sample splits.

In contrast, the deep learning models displayed excellent generalisation properties. The LSTM model maintained stable performance across training and validation sets, and BERT's fine-tuning approach, leveraging pre-trained contextual embeddings, contributed to its exceptional robustness even on unseen data.

## Class Imbalance and Bias Mitigation

Although the original dataset had a slight imbalance (22,851 fake vs 21,416 true), this did not significantly affect model performance due to the relatively even distribution and the use of robust evaluation metrics such as F1-Score, PR-AUC, and ROC-AUC. However, metadata bias was identified in the 'subject' field, where fake news was disproportionately associated with vague or politically charged categories like ‘left-news’ and ‘Government News’. Recognising this, care was taken to either exclude or regularise metadata features to prevent the model from learning superficial correlations.

## Contribution to Existing Literature

The findings of this dissertation align with and extend existing research on fake news detection. Previous studies have shown that deep learning, particularly transformer-based architectures, offers superior performance due to their capacity to learn contextual and hierarchical language representations. By integrating extensive feature engineering, semantic enrichment, and robust model evaluation, this work contributes a detailed, reproducible pipeline for fake news detection that bridges traditional NLP practices with state-of-the-art deep learning.

## Practical Implications

This project’s outcomes have practical significance for social media platforms, news verification services, and government agencies combating misinformation. The ability of fine-tuned BERT to achieve nearly perfect classification suggests its viability for real-time deployment in fact-checking systems. Moreover, the interpretability of engineered features provides transparency that can assist human moderators and journalists in understanding why specific articles are flagged.

## Limitations

Despite strong performance, this project has a few limitations. Firstly, the dataset—although comprehensive—is static and limited to a specific timeframe (primarily 2016–2017). Fake news evolves rapidly, and newer content may contain different linguistic styles or misinformation tactics. Secondly, some advanced models like BERT demand high computational resources, which could limit their real-time applicability in resource-constrained environments.

Finally, while this study focused on textual features, fake news detection can benefit from multimodal approaches that include image analysis, user metadata, and social propagation patterns, which were beyond the scope of this work.

**CHAPTER 7: FUTURE WORK AND CONCLUSION**

# Future Work

While the current study demonstrates the efficacy of both traditional and deep learning models for fake news detection, several promising directions remain open for future exploration:

## Integration of Multimodal Features

This project focused solely on textual content for classification. However, fake news often includes images, videos, and user comments to enhance credibility or virality. Future work could incorporate multimodal learning approaches, combining Natural Language Processing (NLP) with computer vision techniques to analyse accompanying visuals. This would allow a more holistic understanding of misinformation, especially on platforms like Twitter or Facebook, where content is rarely text-only.

## Temporal Dynamics and Event-Aware Detection

Fake news patterns often shift around major political or social events. Developing models that account for temporal context—such as detecting misinformation surges before elections or during crises—would improve adaptability and accuracy over time. Techniques like time-series modelling or recurrent event detection can help address this dynamic nature.

## Cross-Domain and Cross-Language Generalisation

The current dataset is restricted to English-language news with a specific US political bias. Expanding to multilingual datasets and applying domain adaptation techniques could improve model robustness across different cultural, political, and linguistic contexts. Leveraging multilingual BERT (mBERT) or XLM-R models could facilitate this generalisation.

## Explainable AI (XAI) for Trustworthy Decisions

Although BERT achieved exceptional performance, its "black-box" nature can hinder user trust. Future work should focus on integrating explainable AI methods—such as LIME, SHAP, or attention visualisation—to highlight key phrases or patterns contributing to classification. This would increase transparency and trust in automated systems, particularly in high-stakes applications.

## Real-Time Implementation and Scalability

BERT and other transformer models, while powerful, are computationally intensive. Future research could explore efficient deployment strategies such as model quantisation, distillation (e.g., DistilBERT), or cloud-based inference to enable real-time fact-checking pipelines. Scaling these systems to handle high volumes of streaming data remains an open engineering challenge.

## User Behaviour and Propagation Analysis

Another promising area involves analysing how users interact with fake news—such as retweets, likes, or replies. Combining linguistic cues with social network data could enable detection based not only on content but also on dissemination patterns. Graph neural networks (GNNs) and community detection algorithms offer strong potential for such hybrid approaches.

## Conclusion

This dissertation set out to investigate the classification of fake news using a combination of natural language processing, feature engineering, and machine learning techniques. Beginning with a robust dataset of over 44,000 news articles from Kaggle, an extensive series of preprocessing steps—including lemmatisation, stopword removal, sentiment scoring, and readability metrics—was applied to ensure the quality and depth of linguistic features.

A diverse range of models was then trained and evaluated, ranging from baseline Logistic Regression and Random Forest classifiers to advanced architectures such as LSTM and BERT. The evaluation metrics—Accuracy, F1 Score, ROC-AUC, and PR-AUC—consistently highlighted BERT as the most accurate and generalisable model, achieving near-perfect performance across all metrics. The results not only confirm the strength of transformer-based models in understanding complex linguistic patterns but also demonstrate the critical value of thoughtful feature engineering in NLP-based classification tasks.

Importantly, the study went beyond accuracy metrics by incorporating visualisation tools (e.g., word clouds, sentiment plots, network graphs, dimensionality reduction) to offer interpretability and insight into linguistic differences between fake and real news. Furthermore, the examination of metadata biases and the treatment of class imbalance ensured ethical modelling practices were upheld.

Ultimately, this research contributes a reproducible and well-rounded framework for fake news detection and offers strong foundations for future research in more dynamic, multilingual, or multimodal contexts. As misinformation continues to pose challenges in journalism, public health, and democratic processes, the development of accurate and trustworthy AI-based detection systems becomes not only a technical goal but a societal imperative.

# References

* Bosworth, M. L. et al., 2023. Ethnic differences in COVID‑19 mortality in the second and third waves of the pandemic in England during the vaccine rollout: a retrospective, population‑based cohort study. *BMC Medicine,* 21(13).
* Bender, E. M., Gebru, T., McMillan-Major, A. & Shmitchell, S., 2021. On the dangers of stochastic parrots: Can language models be too big?. *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency.*
* Brennen, J. J. S., Simon, F. M., Howard, P. N. & Nielsen, R. K., 2020. *Types, Sources, and Claims of COVID-19 Misinformation,* Oxford: Reuters Institute for the Study of Journalism, University of Oxford.
* Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K., 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint.*
* Golbeck, J., Mauriello, M. & Auxier, B., 2018. Fake news vs satire: A dataset and analysis. *Proceedings of the 10th ACM Conference on Web Science.*
* Goodfellow, L., Leeuwen, E. . v. & Eggo, R. M., 2024. COVID‑19 inequalities in England: a mathematical modelling study of transmission risk and clinical vulnerability by socioeconomic status. *BMC Medicine,* 22(162).
* Joiner, A., McFarlane, C., Rella, L. & Uriarte-Ruiz, M., 2024. Problematising density: COVID-19, the crowd, and urban life. *Social & Cultural Geography,* 25(2), p. 181–198.
* Kaliyar, R. K., Goswami, A. & Narang, P., 2021. FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. *Multimedia Tools and Applications,* Volume 80, p. 11765–11788.
* Lazer, D. M. J., Baum, M. A., Benkler, Y. & Berinsky, A. J., 2018. The science of fake news. *Science,* 359(6380), pp. 1094-1096.
* Public Health England, 2020. *Disparities in the Risk and Outcomes of COVID - 19,* London: Public Health England.
* Raza, S. & Ding, C., 2022. Fake news detection based on news content and social contexts: A transformer-based approach. *International Journal of Data Science and Analytics,* Volume 13, p. 335–362.
* Sharma, S., Patwa, P. & Guptha, V., 2020. Fighting an infodemic: COVID-19 fake news dataset. *Journal of Information Processing Systems,* p. 1123–1139.
* Shu, K. et al., 2017. Fake news detection on social media: A data mining perspective. *arXiv preprint.*
* Vosoughi, S., Roy, D. & Aral, S., 2018. The spread of true and false news online. *Science,* 359(6380), p. 1146–1151.
* Wang, W. Y., 2017. "Liar, liar pants on fire": A new benchmark dataset for fake news detection. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017),* p. 422–426.
* Zhou, X. & Zafarani, R., 2020. *A survey of fake news: Fundamental theories, detection methods, and opportunities,* Syracuse University, USA: arXiv preprint.

|  |  |
| --- | --- |
| APPENDIX | |
| Dataset Files | [Fake.csv](https://uelac-my.sharepoint.com/:x:/g/personal/u2642385_uel_ac_uk/ETjYwVVelQxGnN2PD1GBSY4BUeFOww82VkPed70zpVkZ5Q?e=LMO6Ox) [True.csv](https://uelac-my.sharepoint.com/:x:/g/personal/u2642385_uel_ac_uk/ERVPhG-Zz0VNukGnrAmmZd8BORjiOKaXk7XqWPISUk-g4A?e=YVvD6S) |
| Dataset Source URL | <https://www.kaggle.com/code/therealsampat/fake-news-detection> |
| PythonCode | [ValidationModelsForFakeNewsDetection.ipynb](https://uelac-my.sharepoint.com/:u:/g/personal/u2642385_uel_ac_uk/EZQvbPYN2M5HuRpFNvibp7AByWxDTzv1LJUUCm0eBcYH9A?e=UDnOlb) |
| [EDAFakeNewsDetection.ipynb](https://uelac-my.sharepoint.com/:u:/g/personal/u2642385_uel_ac_uk/EWKf63lFVthKojAdG_Qo-00BnJGeQIlzXzYXRl-VcsY44A?e=ccdBdE) |
| [BERTModelFakeNewsDetection.ipynb](https://uelac-my.sharepoint.com/:u:/g/personal/u2642385_uel_ac_uk/EXO4Mxis-gBHu2VNhj5x3ykBxQkT34adc-5FjRejDWDijQ?e=OSEH4S) |
| Main Folder | [Fake News Detection project](https://uelac-my.sharepoint.com/:f:/g/personal/u2642385_uel_ac_uk/EqHbE-lI4OpAmmpy7NNlctgB79ZskXTzHxU9K7TL-P9yCA?e=8j5K1X) |
|  |  |