# Phase-1 Submission Template

# Student Name: [ R.Arthi ]

# Register Number: [ 511923106003]

# Institution: Priyadarshini Engineering College

# Department: Electronics and Communication Engineering

# Date of Submission: 24/4/2025

# 1.Problem Statement

*In today's digital age, misinformation and fake news spread rapidly through online platforms, influencing public opinion and decision-making. Traditional methods of fact-checking are not scalable to meet the volume and speed at which content is published. Therefore, there is a critical need for an automated system to detect and mitigate the spread of fake news using Natural Language Processing (NLP).*

# 2.Objective of the Project

*The main objective is to develop an intelligent, automated system capable of:*

*Identifying and classifying fake and real news articles using NLP techniques.*

*Providing insight into the linguistic and contextual features of deceptive content.*

*Assisting fact-checkers, media outlets, and the general public in combating misinformation.*

# 3.Scope of the Project

*Focuses on binary classification: real vs. fake news.*

*Applies to textual news content from online articles, blogs, and social media posts.*

*Includes the exploration of NLP-based feature engineering, deep learning, and traditional machine learning models.*

*Will not cover image or video-based fake content detection.*

**4.Data Sources**

*Fake News Detection Dataset (Kaggle): Contains labeled news articles.*

*Example: Fake and real news dataset*

*LIAR Dataset: Short statements labeled with fine-grained classes.*

*BuzzFeed & Politifact Articles: Fact-checked content from known sources.*

*News APIs (e.g., NewsAPI.org): For collecting current news for testing real-time prediction.*

# 5.High-Level Methodology

*Data Collection & Preprocessing*

*Clean and normalize text: remove punctuation, stop words, lowercasing.*

*Tokenization, lemmatization/stemming.*

*Remove duplicates and irrelevant data.*

*Exploratory Data Analysis (EDA)*

*Visualize word frequencies, n-grams, and sentiment scores.*

*Examine metadata (e.g., authorship, source) if available.*

*Feature Engineering*

*TF-IDF, Word2Vec, GloVe, BERT embeddings.*

*Sentiment analysis, linguistic features (e.g., POS tagging, readability scores).*

*Model Development*

*Traditional ML: Logistic Regression, SVM, Random Forest.*

*Deep Learning: LSTM, Bi-LSTM, GRU.*

*Transformer-based Models: BERT, RoBERTa for contextual understanding.*

*Model Evaluation*

*Use metrics: Accuracy, Precision, Recall, F1-Score, ROC-AUC.*

*Cross-validation to avoid overfitting.*

*Deployment (Optional)*

*Develop a web app using Flask/Django.*

*Integrate real-time API for live prediction.*

# 6.Tools and Technologies

*Programming Language: Python*

*NLP Libraries: NLTK, spaCy, TextBlob, Transformers (HuggingFace)*

*Machine Learning: scikit-learn, XGBoost*

*Deep Learning: TensorFlow, Keras, PyTorch*

*Visualization: Matplotlib, Seaborn, Plotly*

*Deployment: Flask, Streamlit, Docker*

*Data Collection: BeautifulSoup, NewsAPI, Tweepy (for Twitter)*

***7.Team Members***

*1.Arthi.R*

*2 Brundha.K*

*3.Mutharasi.A*