**Project Proposal**

**Title:** Fake News Detection Using Natural Language Processing and Machine Learning

**1. Abstract** This project aims to design, develop, and evaluate a system capable of detecting fake news articles using advanced Natural Language Processing (NLP) techniques and machine learning models. By leveraging diverse datasets and classification algorithms, the system will analyze linguistic patterns and semantic features to classify news as either fake or real.

**2. Objectives**

* Collect and preprocess fake and real news datasets.
* Extract meaningful features using NLP techniques.
* Train and evaluate machine learning models for fake news classification.
* Compare the performance of classical ML, deep learning, and transformer-based models.
* (Optional) Deploy a prototype web application for real-time prediction.

**3. Dataset Gathering Strategy**

* Utilize open-source datasets:
  + LIAR Dataset (short political statements with labels)
  + Twitter-based COVID-19 misinformation datasets
  + Other academic or governmental datasets with credibility annotations
* Combine multiple sources to improve coverage and balance.
* Preprocess text data (tokenization, lemmatization, stop-word removal).
* Annotate with metadata if available (source, date, author).

**4. Methodology** **4.1 Data Collection & Cleaning**

* Gather datasets from multiple repositories.
* Perform preprocessing: remove duplicates, normalize casing and punctuation.

**4.2 Exploratory Data Analysis (EDA)**

* Visualize class balance, word frequency, and sentiment distribution.

**4.3 Feature Engineering**

* Use Bag of Words, TF-IDF, and pre-trained word embeddings (Word2Vec, GloVe).
* Utilize BERT for contextual word embeddings.

**4.4 Model Building**

* Train classical ML models: Logistic Regression, Random Forest, SVM.
* Train deep learning models: LSTM, CNN.
* Fine-tune transformer-based models: BERT.

**4.5 Evaluation**

* Evaluate models using accuracy, precision, recall, F1-score, ROC-AUC.
* Select the best performing model based on these metrics.

**4.6 (Optional) Deployment**

* Build a simple web app using Flask or Streamlit for predictions.

**5. Timeline (7 Weeks)**

| **Week** | **Task** |
| --- | --- |
| Week 1 | Literature review, define scope, collect & clean datasets |
| Week 2 | Data preprocessing, EDA, text normalization |
| Week 3 | Feature engineering (TF-IDF, Word2Vec, BERT embeddings) |
| Week 4 | Train classical ML models (LR, RF, SVM), evaluate |
| Week 5 | Implement deep learning models (LSTM/CNN), fine-tune BERT |
| Week 6 | Model comparison, finalize evaluation, write results |
| Week 7 | Write final report, prepare presentation, optional deployment |

**6. Expected Outcomes**

* A comparative analysis of fake news detection models.
* A working classifier capable of distinguishing fake from real news.
* (Optional) A demo web application for real-time predictions.

**7. References**

* LIAR Dataset by William Yang Wang
* Relevant papers and online resources on fake news detection and NLP