FAKE NEWS DETECTION USING NLP

Innovation(Phase 2)

Models we have selected for the project:

- ✓Logistic Regression
- ✓Random Forest
- ✓Neural Networks
- ✓Passive-Aggressive Classifier

MODEL 1: Logistic regression (Effective & Probabilistic algorithm)

- ➢ Logistic Regression is a simple yet effective algorithm for binary classification.

STEP 1 : Initialize

- Initialize model parameters(weights and bias).

STEP 2 : Data Preprocessing

- Preprocess the text including cleaning, tokenization, stopword removal and feature extraction (e.g., TF-IDF).

STEP 3 : Training

- Compute the weighted sum of input features.
- Apply the sigmoid function to the weighted sum to get probabilities.
- Use a loss function (e.g., binary cross-entropy) to measure the error between predicted and actual labels.
- Update model parameters using gradient descent to minimize the loss.

STEP 4 : Prediction

- For a new text sample, preprocess it.
- Compute the weighted sum and apply the sigmoid function.
- Threshold the probability to classify as fake or real news.

MODEL 2: Random Forest

➢ Random Forest is an ensemble learning algorithm that combines multiple decision trees to improve classification performance.

STEP 1: Initialization

- Define the number of decision trees (n_estimators) and other hyperparameters for the Random Forest.

STEP 2: Data Preprocessing

- Preprocess the text data, including cleaning, tokenization, stopword removal, and feature extraction (e.g., TF-IDF or word embeddings).

STEP 3: Training

- Create a random subset (bootstrap sample) of the training data.
- Build multiple decision trees, each trained on a different bootstrap sample.
- At each node of a decision tree, randomly select a subset of features for splitting.
- Use majority voting (classification) or averaging (regression) to make predictions across all decision trees.

STEP 4: Prediction

- For a new text sample, preprocess it.
- Pass the sample through each decision tree in the forest to obtain individual predictions.
- Aggregate the predictions (e.g., by majority voting) to make the final classification.

MODEL 3: Neural Networks (Feedforward Neural Network)

- A Feedforward Neural Network is a type of artificial neural network that consists of an input layer, hidden layers, and an output layer.

STEP 1 : Initialization

- Define the architecture of the neural network, including the number of layers, number of neurons in each layer, activation functions, and optimization algorithm.

STEP 2 : **Data Preprocessing**

- Preprocess the text data, including tokenization, padding, and converting tokens to input embeddings (e.g., word embeddings or one-hot encoding).

STEP 3 : **Training**

- Feed the preprocessed text data into the neural network.
- Forward propagation: Compute activations in each layer by applying activation functions.
- Compute the loss (e.g., cross-entropy) between predicted and actual labels.
- Backpropagation: Calculate gradients of the loss with respect to the model's parameters.
- Update model weights using an optimization algorithm (e.g., gradient descent) to minimize the loss.
- Repeat these steps for multiple epochs until convergence.

STEP 4 : **Prediction**

- For a new text sample, preprocess it and convert it into the neural network's input format.
- Pass the sample through the trained neural network to obtain predictions from the output layer.

MODEL 4: Passive Aggressive Classifier

- ➢ The Passive Aggressive Classifier is a type of online learning algorithm suitable for text classification tasks.

STEP 1 : **Initialization**

- Define hyperparameters, such as the aggressiveness parameter (C).

STEP 2 : **Data Preprocessing**

- Preprocess the text data, including cleaning, tokenization, stopword removal, and feature extraction (e.g., TF-IDF or word embeddings).

STEP 3 : **Training**

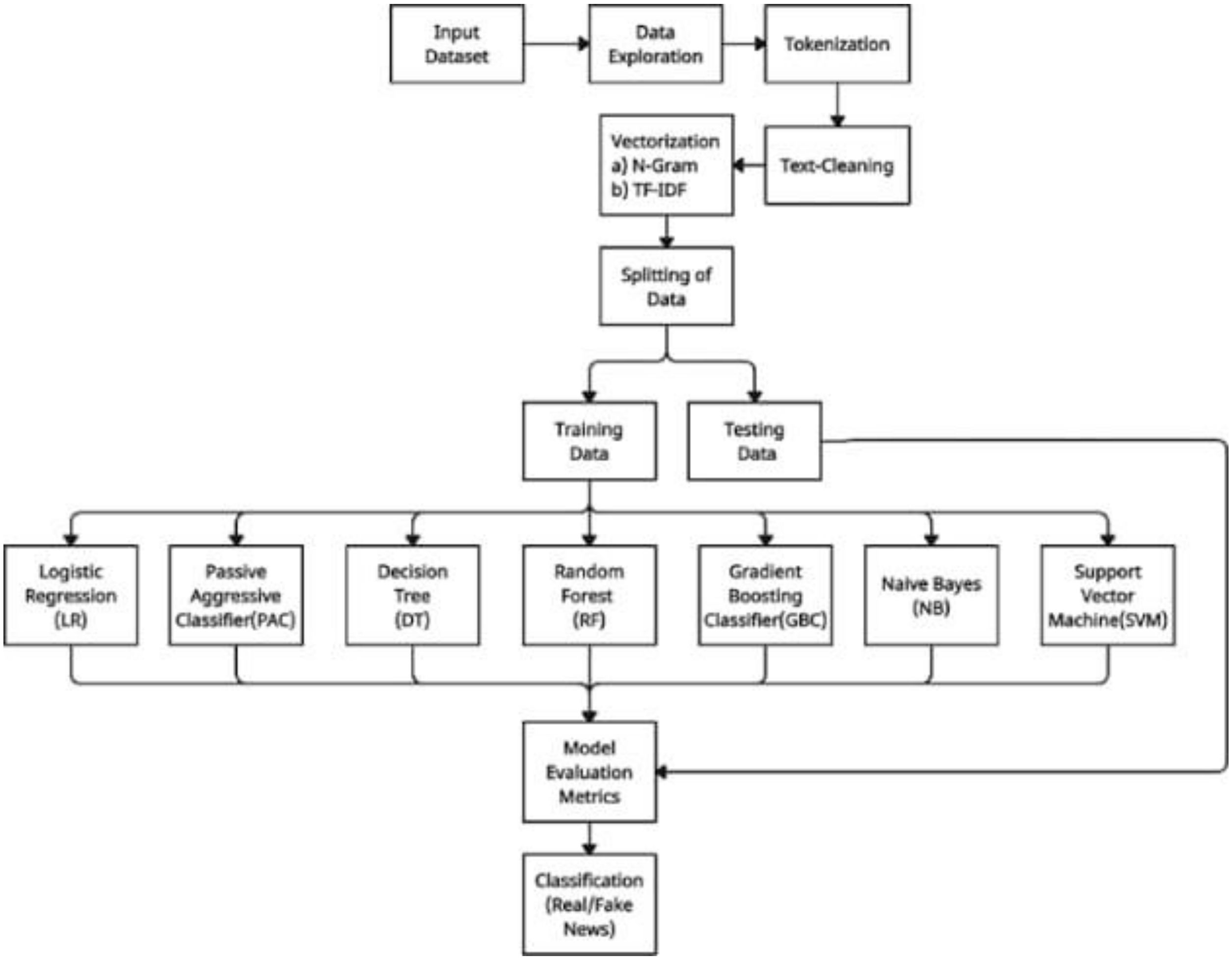
For each training sample:

- Compute the prediction based on the current model.
- Calculate the loss or error between the predicted label and the true label.
- Update the model parameters to minimize the loss, with an emphasis on making the correct prediction.
- Adjust the aggressiveness parameter (C) to control the learning rate.

STEP 4 : **Prediction**

- For a new text sample, preprocess it.
- Use the trained model to predict the label based on the input data.

FLOWCHART



CONCLUSION

- In conclusion, the application of Natural Language Processing (NLP) algorithms for fake news detection represents a promising avenue in the battle against misinformation. These algorithms leverage linguistic patterns, semantic analysis, and machine learning to identify potentially deceptive content. NLP algorithms are not infallible and can encounter challenges in handling subtle misinformation, satire, and context-dependent content. Therefore, continuous research and development are essential to enhance their accuracy and adaptability.