Fake news detection using NLP

Design Thinking :

**1. Data source :**

- Fake and real news dataset. Text classification dataset. Set of fake and real news for binary classification.

- Face Mask Detection. Computer vision. Object detection dataset for identifying masks on human faces.

- Brain MRI Images for Brain Tumor Detection. Computer vision. Image classification dataset on medical data.

**2. Data Preprocessing :**

- Gather and clean the dataset, removing any irrelevant or duplicate information.

- Tokenize the text data, splitting it into words or subword tokens.

- If your textual data has missing values, decide whether to impute missing values or remove rows with missing data, depending on the dataset's size and significance of the missing data.

**3. Feature Extraction :**

- Utilize techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings to convert text into numerical features.

- TF-IDF is a statistical measure used to evaluate the importance of a word in a document relative to a collection of documents (corpus).

- It assigns a weight to each word based on its frequency within the document and its rarity across the corpus.

- TF-IDF is useful for text classification, information retrieval, and document similarity tasks.

e.g (Word2Vec, GloVe)

- Consider incorporating metadata like author information or publication date if available.

**4. Model Selection :**

- Experiment with various machine learning model Random Forest algorithm, or more advanced models LSTM or Transformer-based models.

**Random Forest:**

Suitability:

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. It's highly suitable for classification tasks and can handle a variety of data types and feature complexities.

Advantages:

Robust to overfitting, capable of handling both numerical and categorical data, and provides feature importance scores.

e.g (BERT, GPT-3).

- Fine-tune hyperparameters to optimize model performance.

**5. Model Training :**

- Train the selected model(s) on the training data.

- Monitor metrics like accuracy, precision, recall, and F1-score during training.

- Fit the chosen model to the training data using the fit method. This step involves adjusting the model's parameters to make it learn from the training data.

**6. Evaluation :**

1. Accuracy:

- Accuracy measures the overall correctness of the model's predictions. It's the ratio of correctly predicted instances to the total instances in the test set.

2. Precision:

- Precision measures the proportion of true positive predictions among all positive predictions made by the model. It's especially useful when false positives are costly or undesirable.

3. Recall (Sensitivity or True Positive Rate):

- Recall measures the proportion of true positive predictions among all actual positive instances. It's useful when you want to ensure that all positive cases are captured.

4. F1-Score:

- The F1-score is the harmonic mean of precision and recall. It provides a balanced measure of a model's accuracy, especially when there is an imbalance between the classes.

5. ROC-AUC (Receiver Operating Characteristic - Area Under the Curve):

- ROC-AUC is used for binary classification problems and measures the model's ability to distinguish between the positive and negative classes. It's often used when you want to assess the trade-off between true positive rate (sensitivity) and false positive rate.

- Assess the model's performance using the validation set.

- Tweak the model and hyperparameters as needed.