Report Structure

Preprocessing and TF-IDF Vectorization

- **Lemmatization and Text Cleaning**: Applied lemmatization to normalize the text, along with lowercasing, removing URLs, emojis, punctuation, and stopwords.

- **TF-IDF Vectorization**: Utilized TfidfVectorizer with a maximum of 5000 features to transform the cleaned text data into TF-IDF vectors.

Model Training and Hyperparameter Details

1. **DNN**

- Best Hyperparameters:

Best trial accuracy: 0.9306603773584906
Best trial parameters:
lr: 6.567124316089822e-05
batch_size: 32
n_layers: 4
n_units_l0: 288
n_units_l1: 293
n_units_l2: 261
n_units_l3: 507
dropout_rate: 0.4906446480577282
optimizer: Adam

2. **CNN**

- Best Hyperparameters:

Best trial:

{'lr': 0.0014350664903037842, 'batch_size': 64, 'num_filters': 100, 'dropout_rate': 0.12954789773191497, 'optimizer': 'Adam', 'filter_sizes': '(2, 3, 4)', 'padding': 'same'}

Best CNN model saved as 'best_cnn_model.pth'

3. **LSTM**

- Best Hyperparameters:

{'hidden_dim': 184, 'num_layers': 2, 'dropout': 0.4427079694783371, 'lr': 0.001357000806982032, 'batch_size': 16, 'optimizer': 'SGD'}

Model Saving and Sharing

- Used joblib to the TF-IDF vectorizer.

- Used PyTorch to save the models by ‘model_name.pth’.

- Shared via Google Drive: [Model Files]
https://drive.google.com/drive/folders/1HRA-K1-OAOYa912lX6EvtU-DjNzIVjfp?usp=drive_link