Text Classification 2

Import Necessary Libraries

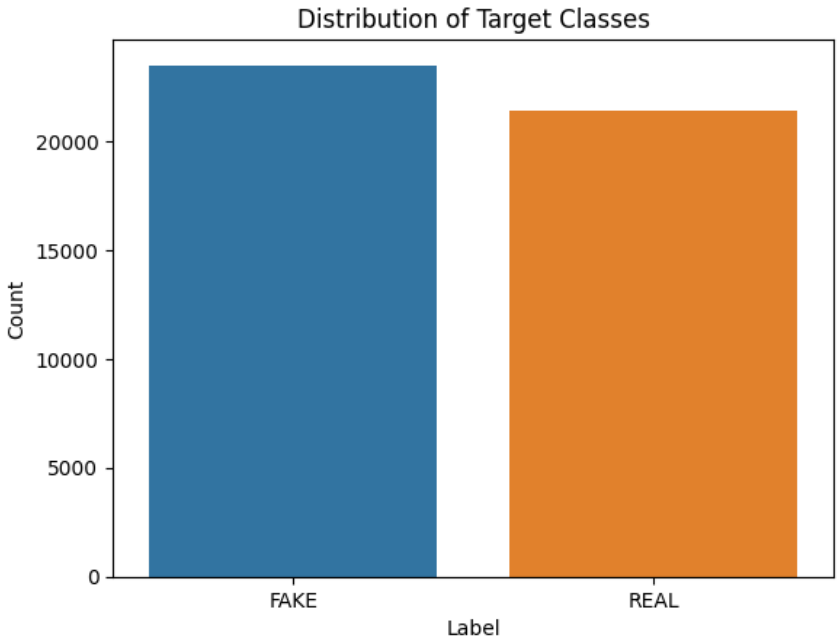
```
import tensorflow as tf
from tensorflow.keras import datasets, layers, models
from tensorflow.keras.preprocessing.text import Tokenizer
import pandas as pd
import seaborn as sb
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.layers import Embedding, LSTM, Dense
from tensorflow.keras.models import Sequential
import matplotlib.pyplot as plt
import numpy as np
from sklearn.preprocessing import LabelEncoder
from tensorflow.keras.callbacks import EarlyStopping
from sklearn.metrics import classification_report
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
```

For this assignment, I'll be using real news and fake news dataset from Kaggle.

```
# Load the fake news dataset
df_fake = pd.read_csv('Fake.csv')
df_fake['label'] = 'FAKE'

# Load the real news dataset
df_real = pd.read_csv('True.csv')
df_real['label'] = 'REAL'

# Concatenate the two datasets into a single DataFrame
df = pd.concat([df_fake, df_real])

# Split the data into train and test sets
from sklearn.model_selection import train_test_split

X = df['text']
y = df['label']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, stratify=y, random_state=42)
```

```
# Load the data
df = pd.read_csv('Fake.csv')
df['label'] = 'FAKE'
df2 = pd.read_csv('True.csv')
df2['label'] = 'REAL'
df = pd.concat([df, df2]).reset_index(drop=True)

from sklearn.model_selection import train_test_split

x = df.text
y = df.label

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, stratify=y, random_state=42)

sb.countplot(x='label', data=df)
plt.title('Distribution of Target Classes')
plt.xlabel('Label')
plt.ylabel('Count')
plt.show()
```

```
print(df.describe())
```




```
                                                  title   text  \
count                                             44898  44898
unique                                            38729  38646
top     Factbox: Trump fills top jobs for his administ...
freq                                                 14    627

             subject               date  label
count          44898              44898  44898
unique             8               2397      2
top     politicsNews  December 20, 2017    FAKE
freq           11272                182  23481
```

Double-click (or enter) to edit

```
# divide into train and test sets
np.random.seed(1234)
i = np.random.rand(len(df)) < 0.8
train = df[i]
test = df[~i]
print("Train data size: ", train.shape)
print("Test data size: ", test.shape)
```

```
Train data size:  (35941, 5)
Test data size:  (8957, 5)
```

Double-click (or enter) to edit

```
num_labels = 2
vocab_size = 25000

tokenizer = Tokenizer(num_words=vocab_size)
tokenizer.fit_on_texts(train.text)

x_train = tokenizer.texts_to_matrix(train.text, mode='tfidf')
x_test = tokenizer.texts_to_matrix(test.text, mode='tfidf')


# create a validation set
x_val = x_train[:10000]
partial_x_train = x_train[5000:]
```

```
y_val = y_train[:10000]
partial_y_train = y_train[5000:]


model = models.Sequential()

model.add(layers.Dense(1, input_dim=x_train.shape[1], activation='relu'))
model.add(layers.Dense(num_labels, activation='softmax'))

model.compile(optimizer='rmsprop',
              loss='binary_crossentropy',
              metrics=['accuracy'])

history = model.fit(x_train,
                    y_train,
                    epochs=10,
                    batch_size=128)



score = model.evaluate(x_test, y_test, 128, verbose =1)
print('Accuracy: ', score[1])
```

✓ 0s completed at 6:14 PM