NLP Project

FCAI-HU

Team Number: 5

Team Members Names & IDs:

Team Member Name	Team Member ID	
احمد مصطفى اسماعيل علام	**201900103**	
مصطفى عصام عبدالفتاح ابوشامه	**201900824**	
محمد عبد الرحيم ابراهيم محمد	**201900698**	
رنا عادل فرج	**201900306**	

Dataset Details

Dataset Name: Medical Text Dataset - Cancer Doc Classification.

Dataset Link:
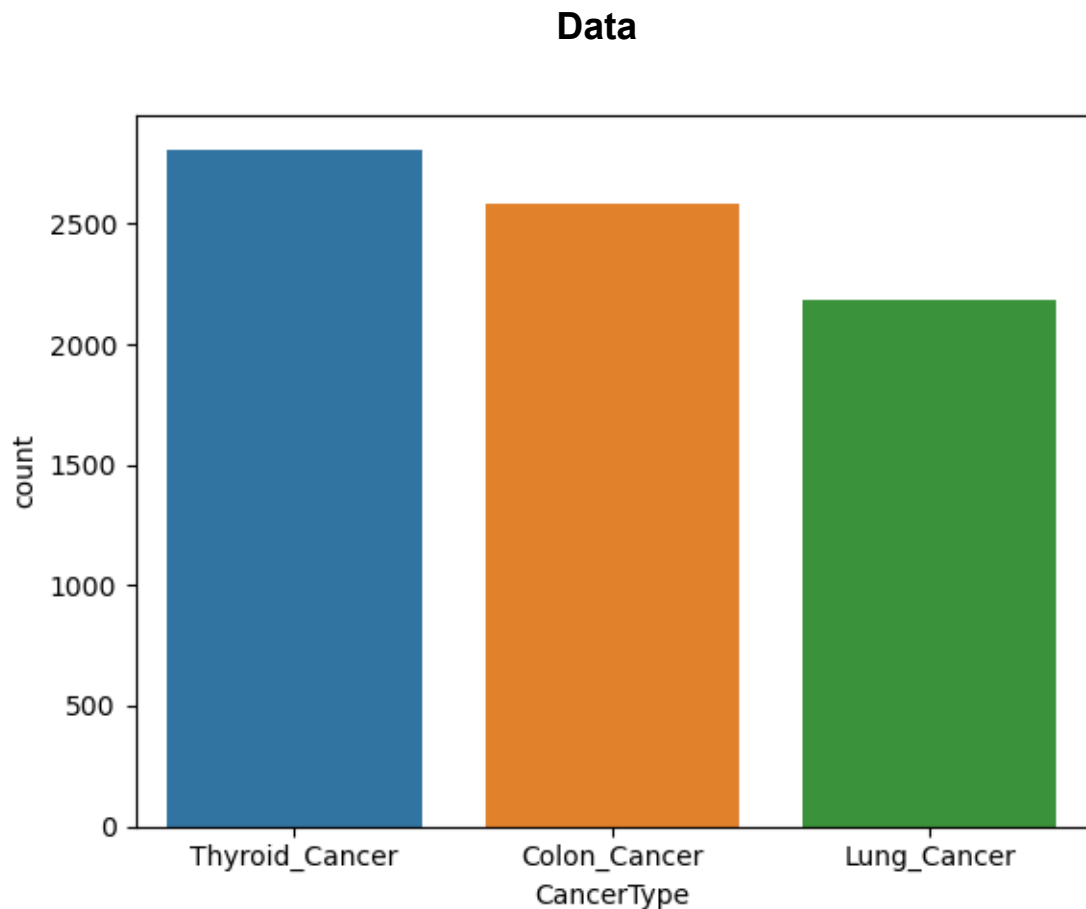
[Medical Text Dataset -Cancer Doc Classification | Kaggle](#)

Number of Samples: 7570
[Train : Test] Ratio -> 6056 text : 1514 text = **80% : 20%**

Number of Classes: 3
Classes Labels: Thyroid Cancer, Lung Cancer, Colon Cancer

Analysis & Distribution Data

Data

This graph shows that data is <u>balanced</u>

Model

```python
model = Sequential()

model.add(Embedding(max_words, 100, input_length = max_Text_length))
model.add(LSTM(128))
# model.add(BatchNormalization())
model.add(Dropout(0.25))

model.add(Flatten())
model.add(Dense(64, activation = 'relu'))
model.add(Dropout(0.25))

model.add(Dense(3, activation = 'softmax'))

model.compile(optimizer = 'adam', loss = 'sparse_categorical_crossentropy', metrics = ['accuracy'])
history = model.fit(X_train, y_train, validation_data = (X_test, y_test), epochs = 10)

y_pred = model.predict(X_test)

model.summary()
```
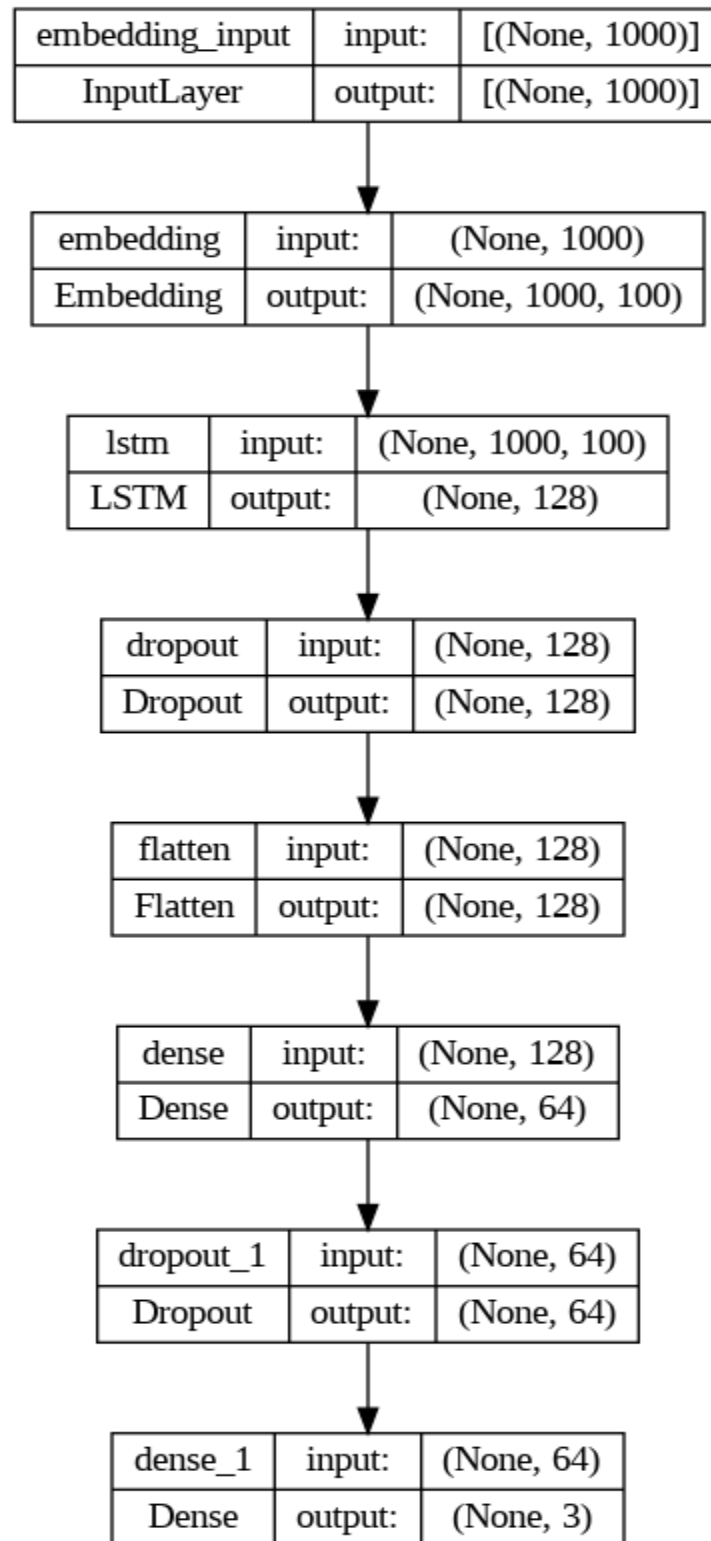
Block Diagram:

embedding_input	input:	[(None, 1000)]
InputLayer	output:	[(None, 1000)]

embedding	input:	(None, 1000)
Embedding	output:	(None, 1000, 100)

lstm	input:	(None, 1000, 100)
LSTM	output:	(None, 128)

dropout	input:	(None, 128)
Dropout	output:	(None, 128)

flatten	input:	(None, 128)
Flatten	output:	(None, 128)

dense	input:	(None, 128)
Dense	output:	(None, 64)

dropout_1	input:	(None, 64)
Dropout	output:	(None, 64)

dense_1	input:	(None, 64)
Dense	output:	(None, 3)

Steps of our Algorithm:

Preprocessing: We removed all stopwords, We removed words of length equal or less than 2, We have used Regular Expression to made all data sets without numbers or strange symbols and make them in lowercase letters, We use Lemmtizations to make every word in base dictionary form, We have made all words contain 1,000 words of fixed length to prepare data before feeding them into the network.

LSTM Architecture: The LSTM architecture is defined and consists of 128 Units followed by two layers, and we add batch normalization and dropout (type of regularization) to avoid overfitting.

Training: The LSTM model is trained on a dataset of 6056 text belonging to three different types of Cancer (Thyroid, Lung, Colon).

Testing: The final step is testing the model on new unseen 1514 text to classify them into one of the three Cancer types.

Results: Finally, we get the Accuracy = 96.24%

Hyper-parameters:

```
numOf(LSTM)Layers = 1
numOf(units_in_LSTM) = 128
numOfDenseLayers = 2 & Activation_functhion = 'relu', 'softmax' for output
numOf(units_in_FirstDenseLayer) = 64
optimizer = 'adam'
loss = 'sparse_categorical_crossentropy'
epochs = 10
batch_size = 32 (Defualt)
maxWord_in_Tokenizer = 25000
maxTextLength = 1000
```

We have used LSTM once, and another once we have used Bidirectional with LSTM.

We have changed the hyperparameters in model many times <u>to get the best results.</u>
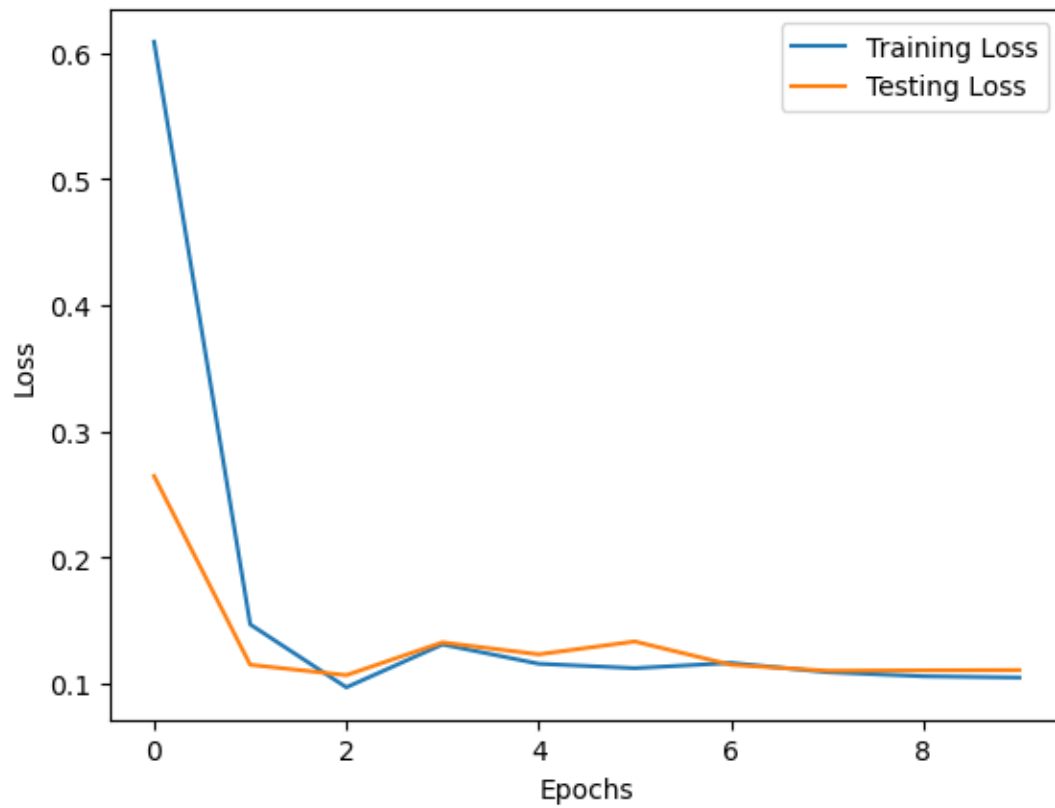
Result Details:

1) Accuracy Graph:
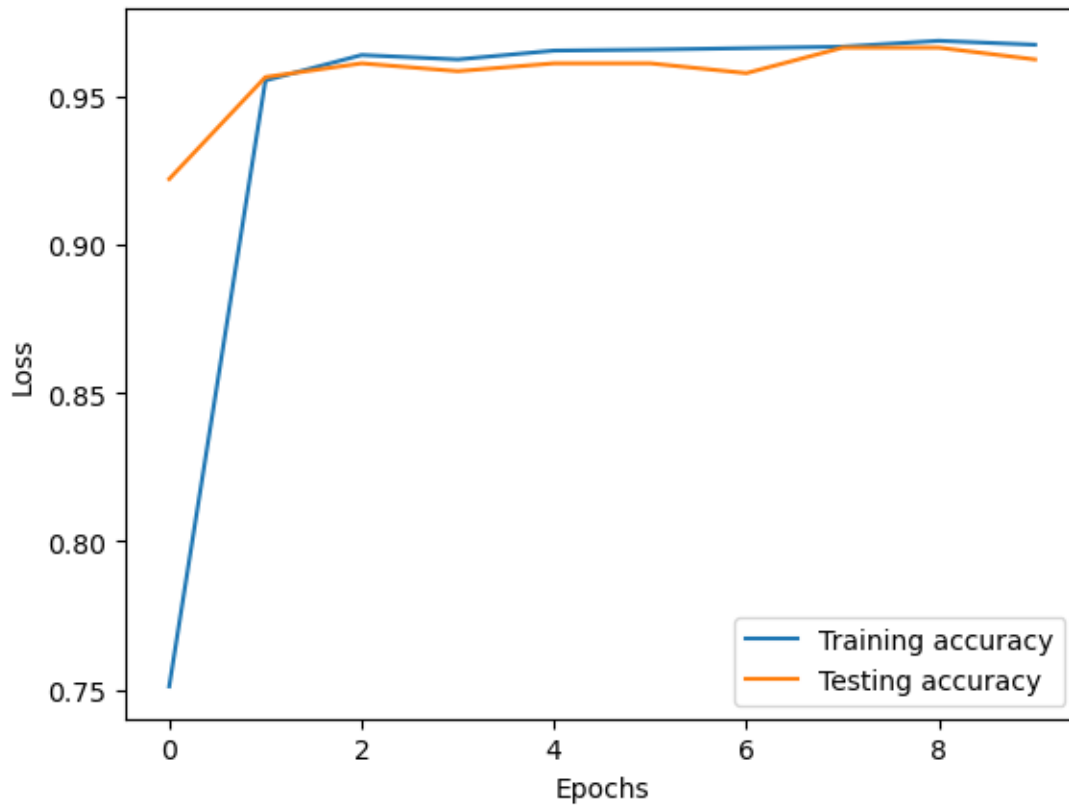
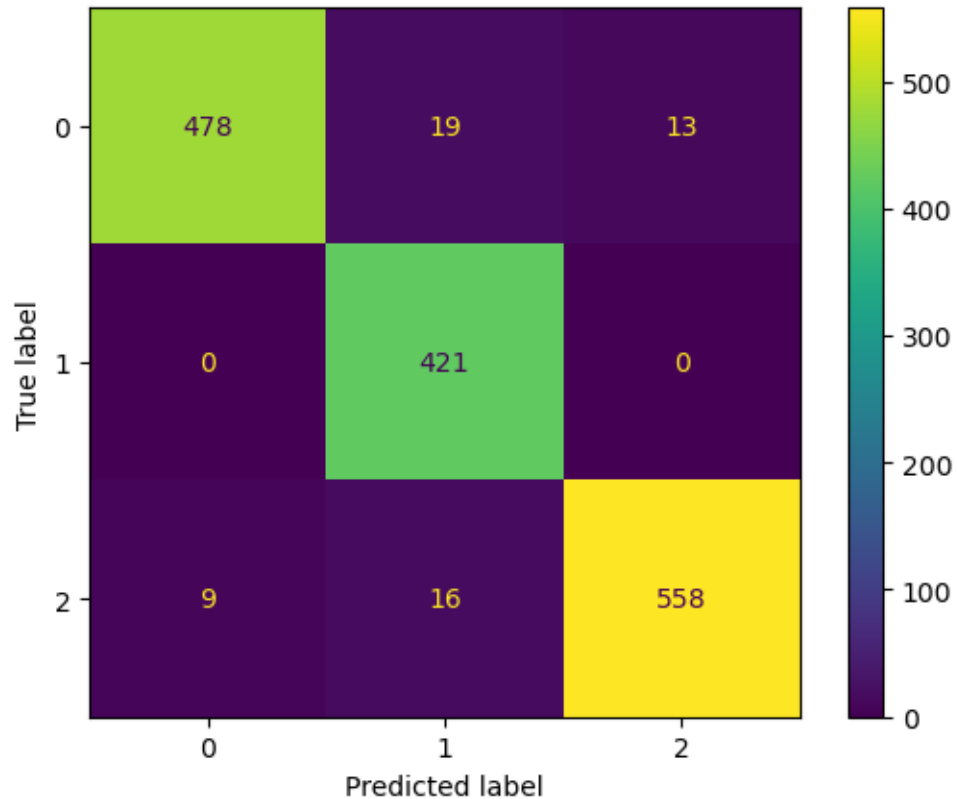**Training Accuracy = 96.73 %**

Validation Accuracy = 96.24 %

2) Loss Graph:

Training Loss = 10.47 %

Validation Loss = 11.07 %

3) Confusion Matrix:

4) Zero One Loss: -> Only 57 of 1514 image

There are <u>1457 of 1514</u> text that Model Predicted Correctly.

....