**\*\*# Title and Description\*\***

A Dockerized FastAPI-based REST service that classifies medical research papers into 'Cancer' or 'Non-Cancer' using a fine-tuned DistilBERT model with LoRA adapters. Deployed on Hugging Face Spaces using Flask and has two end points.

Additionally, it uses out of the box LLM Phi-4 for Disease classification based on existing text.

**\*\*# Demo\*\***

Try it out (https://huggingface.co/spaces/tarunchander/DistillBert). The request to API can be sent through RequestURL.ipynb

**\*\*# High Level Steps\*\***

* Experimented with a sample of 100 records on following models with same hyperparameters:
  + Roberta:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Epoch** | **Training Loss** | **Validation Loss** | **Accuracy** | **F1** |
| 1 | 0.693200 | 0.689373 | 0.550000 | 0.354839 |
| 2 | 0.688000 | 0.686869 | 0.550000 | 0.354839 |
| 3 | 0.682500 | 0.686213 | 0.550000 | 0.354839 |
| 4 | 0.668200 | 0.685930 | 0.550000 | 0.354839 |
| 5 | 0.688800 | 0.685794 | 0.550000 | 0.354839 |

* + Bert:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Epoch** | **Training Loss** | **Validation Loss** | **Accuracy** | **F1** |
| 1 | 0.697300 | 0.713131 | 0.200000 | 0.427500 |
| 2 | 0.705300 | 0.710935 | 0.200000 | 0.445701 |
| 3 | 0.687000 | 0.712721 | 0.250000 | 0.402564 |
| 4 | 0.687400 | 0.713695 | 0.300000 | 0.346154 |
| 5 | 0.687700 | 0.714815 | 0.300000 | 0.304348 |

* + DistillBert:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Epoch** | **Training Loss** | **Validation Loss** | **Accuracy** | **F1** |
| 1 | 0.693100 | 0.687153 | 0.450000 | 0.310345 |
| 2 | 0.680400 | 0.673540 | 0.450000 | 0.475275 |
| 3 | 0.669100 | 0.656694 | 0.700000 | 0.762032 |
| 4 | 0.648100 | 0.644986 | 0.650000 | 0.703804 |
| 5 | 0.636100 | 0.637990 | 0.700000 | 0.721591 |

**Reasons for choosing DistillBert:**

* + Higher accuracy and lower validation on sample Dataset.
  + Smaller model would be efficient to train and reduce inference cost also.
  + As the dataset is small as only 200K tokens need to be trained, choosing 300 mn model will not be add significant value when same can be performed using 16 mn model.

**\*\*# Code Files & Steps\*\***

* **DataPreprocessing.ipynb:** Loaded the all research articles in pipe delimited file extracting information such as ID,Title\_Abstract and Label(Cancer/NonCancer). The data is kept in combined\_file.csv
* **Trainer\_v0.2:**
* Loaded above combined\_file and Used MultiLabelBinarizer for encoding Label column in two classes [0,1].
* Resampled the entire dataset so that Cancer/NonCancer rows are randomly ordered.
* Load Model using AutoModelForSequenceClassification and also tokenizer .
* Split the dataset into train/test
* Tokenize the train and test dataset to get title\_abstract, label\_id, input\_id
* Load the peft model using get\_peft\_model
* Define training parameters and compute metrics.
* Train the model using Trainer
* Predict test dataset using trained model and apply softmax
* Calculate Accuracy,confusion matrix and F1 score.
* Merged peft and saved model locally.
* **app.py:**
  + Use Flask API to create 3 end points:
  + Default end point
  + Multi-Label Classification Output using finetuned model
  + Disease Classification end point for classifying disease using LLM
* **RequestURL.ipynb:** To invoke 2 end points.
* [**Dockerfile**](https://github.com/tarunchander1234/CodeForClassification/blob/main/Dockerfile)**:** To create docker image.
* Pushed code to github
* Pushed docker image to containerize on Hugging space.

**\*\*# Assumption\*\***

- It has been assumed that labels are not noisy and research articles are classified in folders accurately.