* **Training Pipeline Details**

i prepared the training and validation dataset like this :

step 1 : i merged trained and validation set of **Stanford Sentiment Treebank v2 (SST2)** dataset together

step 2 : i merge/mix full **Bengali News Comments** dataset with newly created sst-2 train dataset

step 3 : i merge/mix randomly selected 70% data from **RomanicBanglaSentiment** dataset(that azim labeled) with newly created train dataset and remaining 30% data(1500 samples) goes into my validation set

i forgot to add this line of code in training notebook : **model.save\_weights('model\_checkpoint\_1.h5')** please add this just after the last cell of training notebook to save trained models best weight file for further inference

Check training notebook titled **Train-Romanic bangla-murad-takla-sentiment-analysis-xlmr.ipynb** for training pipeline with proper explanation

Check inference notebook titled **inference-Romanic bangla-murad-takla-sentiment-analysis.ipynb** for demo gpu inference

After fine tuning xlm roberta large i was able to achieve 0.9861 validation accuracy on 16th epoch, please view full train log from **full training log.txt** file

* **How to Reproduce result and generate weight file?**1. Upload all the dataset in kaggle (you will find all the dataset inside the attached zip file and inside **datasets** folder)  
  2. Create new notebook in kaggle and then from file menu select and upload **Train-Romanic bangla-murad-takla-sentiment-analysis-xlmr.ipynb** file  
  3. Select tpu v3-8 hardware  
  4. Add this line of code **model.save\_weights('model\_checkpoint\_1.h5')** at the bottom of the notebook

5. hit the commit button(it will take around 3 hours to finish training) and then you will get **model\_checkpoint\_1.h5** weight file attached as notebooks output  
6. Download **model\_checkpoint\_1.h5** weight file from notebooks output and use that weight file to run **inference-Romanic bangla-murad-takla-sentiment-analysis.ipynb** notebook in order to make inference using the weight file

* **How to improve and make the model more generalized?**   
  I tried many experiments and I think you can play with the code snippet below :   
    
  val = pd.concat([trainData[['text','sentiment']].query('sentiment==1').sample(n=1000, random\_state=0),trainData[['text', 'sentiment']].query('sentiment==0').sample(n=500, random\_state=0)])

val.reset\_index(inplace = **True**)

len(val)

just try to increase sample size from say n = 500 to 700 and 1000 to 2000, this way you will add more data in validation set and still I believe you can achieve >95% validation accuracy with ease,thank you