TEXT CLASSIFICATION USING DEEP LEARNING

GROUP 26: TEAM MEMBERS	STUDENT ID	Email ID
Adorin Kripanand Lucas	300322905	akrip010@uottawa.ca
Daksh Chaudhary	300322404	dchau012@uottawa.ca
Aishwarya Manapuram	300322316	amana022@uottawa.ca

Division of Tasks:

Both part 1 and part 2 were divided amongst the team mates equally. The problem statement and ways to solve the tasks were discussed and understood together over MS Teams. The errors in the code for part 1, issues and changes in scripts for part 2 were resolved by all of us together after referring multiple resources online.

The evaluation score using the similarity values obtained vs the standard values for Pearson correlation was done by all the team mates together. The results and documentation of the whole assignment was discussed and written together.

Part 1: Legal text classification

Below are the AUPR, Precision and Recall values for **pre-trained** Language models after running the code from GITHUB: https://github.com/TheAtticusProject/cuad

RoBERTA-base:
AUPR: 0.426,
Precision at 80% Recall: 0.311,
Precision at 90% Recall: 0.000

RoBERTA-large:
AUPR: 0.482,
Precision at 80% Recall: 0.381,
Precision at 90% Recall: 0.000

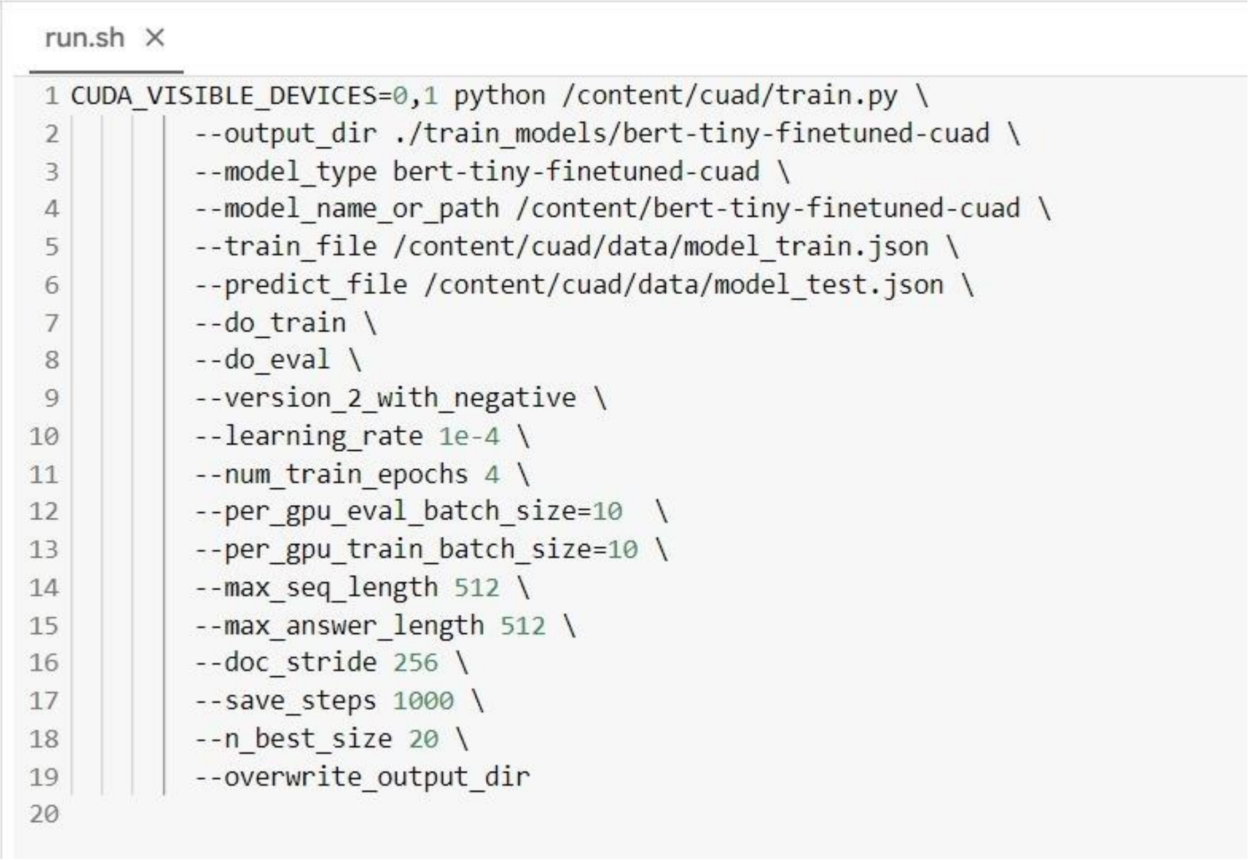
DeBERTA-v2-xlarge:
AUPR: 0.478,
Precision at 80% Recall: 0.440,
Precision at 90% Recall: 0.178

WORKING and EVALUATION:

- We have used the pre-trained models on CUAD from hugging face site (https://huggingface.co/) to train our data (16 training files and 4 test files). So, 20 files in total where 80:20 ratio is maintained.
- We have reduced the number of files because of hardware limitations.
- For every model, the code remains the same except the git cloning part of the model and all the changes are made in the run.sh file.
- For every model, the output_dir, model_type, model_name_or_path is changed. We have modified train_file and predict_file values too.
- Here, instead of the actual number of files (500+), we are inputting out subset json files (16 for train and 4 for test)

Run.sh file for bert-tiny-finetuned-cuad

```
CUDA_VISIBLE_DEVICES=0,1 python /content/cuad/train.py \
      --output_dir ./train_models/bert-tiny-finetuned-cuad \
      --model_type bert-tiny-finetuned-cuad \
      --model_name_or_path /content/bert-tiny-finetuned-cuad \
      --train_file /content/cuad/data/model_train.json \
      --predict_file /content/cuad/data/model_test.json \
      --do_train \
      --do_eval \
      --version_2_with_negative \
      --learning_rate 1e-4 \
      --num_train_epochs 4 \
      --per_gpu_eval_batch_size=10  \
      --per_gpu_train_batch_size=10 \
      --max_seq_length 512 \
      --max_answer_length 512 \
      --doc_stride 256 \
      --save_steps 1000 \
      --n_best_size 20 \
      --overwrite_output_dir
```



```
CUDA_VISIBLE_DEVICES=0,1 python /content/cuad/train.py \
        --output_dir ./train_models/bert-tiny-finetuned-cuad \
        --model_type bert-tiny-finetuned-cuad \
        --model_name_or_path /content/bert-tiny-finetuned-cuad \
        --train_file /content/cuad/data/model_train.json \
        --predict_file /content/cuad/data/model_test.json \
        --do_train \
        --do_eval \
        --version_2_with_negative \
        --learning_rate 1e-4 \
        --num_train_epochs 4 \
        --per_gpu_eval_batch_size=10  \
        --per_gpu_train_batch_size=10 \
        --max_seq_length 512 \
        --max_answer_length 512 \
        --doc_stride 256 \
        --save_steps 1000 \
        --n_best_size 20 \
        --overwrite_output_dir

```

RESULT:
The Below are the AUPR values obtained:
Roberta-base-cuad-finetuned:
AUPR: 0.348

Distilbert-base**-uncased-finetuned-cuad**:
AUPR: 0.008

Bert-Tiny-Finetuned-Cuad:
AUPR: 0.004

Bert-Base-uncased-squad-v1:
AUPR: 0.08

NOTE: Please find the code attached in NLP_Assignment2_Task1 in Task 1 - Code and Results folder.

Part 2: Training sentence similarity models

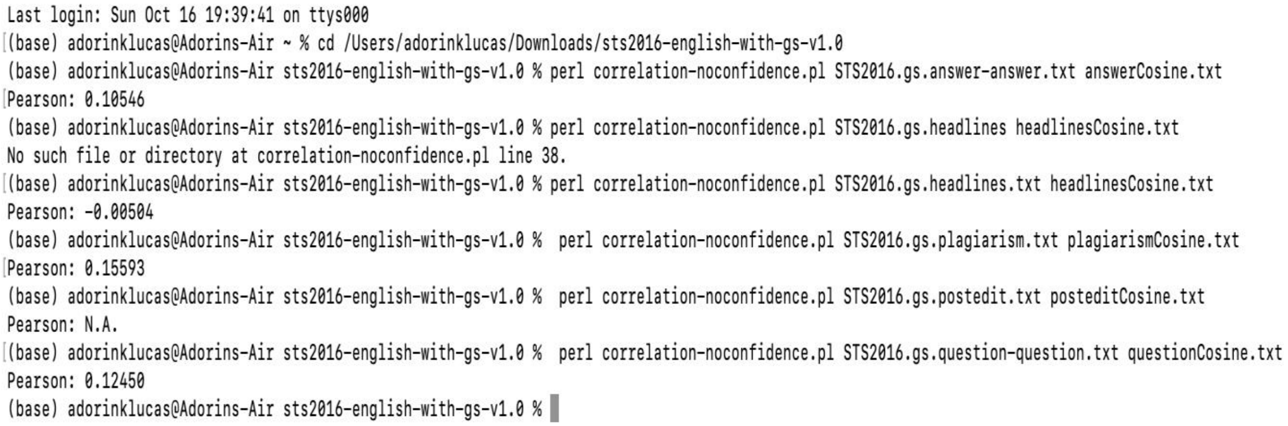
5 text files (STS2016.input.answer-answer.txt, STS2016.input.headlines.txt, STS2016.input.plagiarism.txt, STS2016.input.postediting.txt, STS2016.input.question-question.txt) are taken into consideration for Sentence Embedding Evaluations.

The below table are the **results of Task 2 from Assignment 1**. From the below 5 models, we chose to fine tune the **SBERT** model.

Datasets	**Doc2vec**	**SBERT**	**InferSent**	**Universal Sentence Enoder**	**all-mpnet-base-v2**	**Best Score**
STS2016.input.answer-answer.txt	0.03546	0.10546	0.36618	0.52688	0.54591	0.54591
STS2016.input.headlines.txt	0.38985	-0.00504	0.38099	0.67063	0.36420	0.67063
STS2016.input.plagiarism.txt	0.01401	0.15593	0.39360	N.A.	0.05664	0.39360
STS2016.input.postediting.txt	-0.10898	NA	0.14023	0.74557	0.37308	0.74557
STS2016.input.question-question.txt	0.03168	0.12450	0.18692	0.63687	0.23106	0.63687

Resources referred: https://www.sbert.net, https://www.analyticsvidhya.com/blog/2020/08/top-4-sentence-embedding-techniques-using-python/

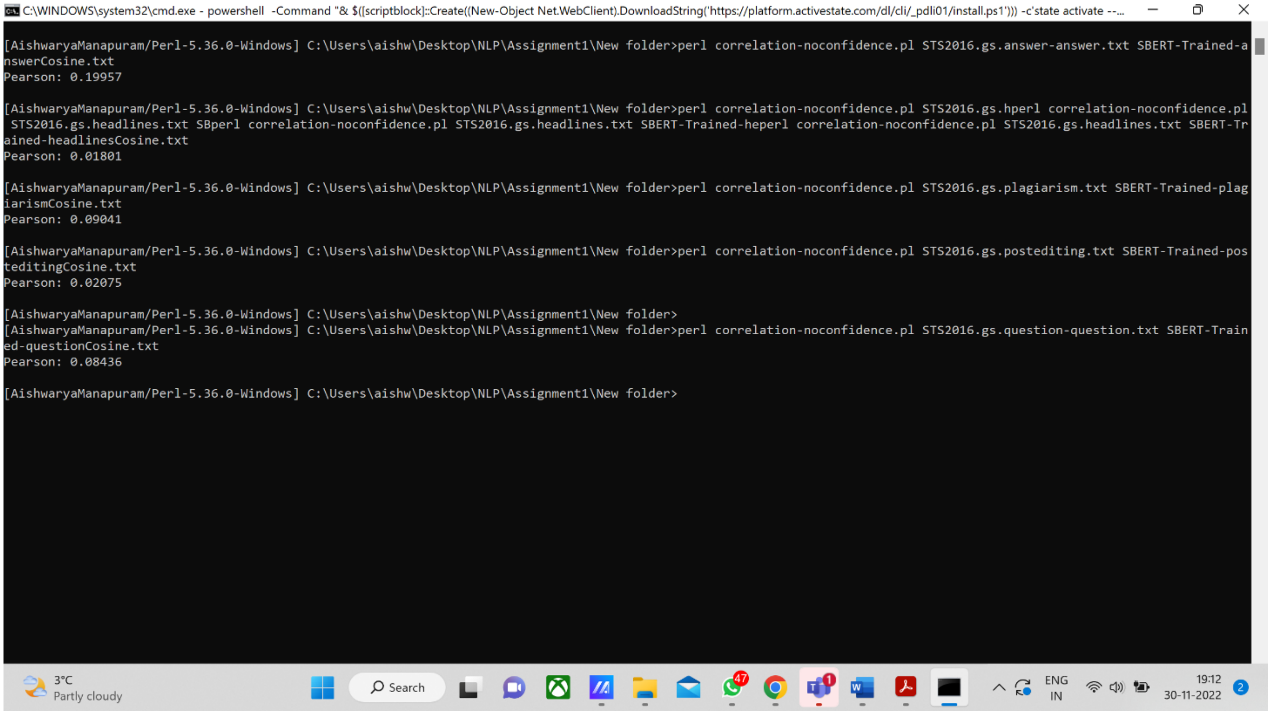
SBERT Values from Assignment 1



```
Last login: Sun Oct 16 19:39:41 on ttys000
(base) adorinklucas@Adorins-Air ~ % cd /Users/adorinklucas/Downloads/sts2016-english-with-gs-v1.0
(base) adorinklucas@Adorins-Air sts2016-english-with-gs-v1.0 % perl correlation-noconfidence.pl STS2016.gs.answer-answer.txt answerCosine.txt
Pearson: 0.10546
(base) adorinklucas@Adorins-Air sts2016-english-with-gs-v1.0 % perl correlation-noconfidence.pl STS2016.gs.headlines headlinesCosine.txt
No such file or directory at correlation-noconfidence.pl line 38.
(base) adorinklucas@Adorins-Air sts2016-english-with-gs-v1.0 % perl correlation-noconfidence.pl STS2016.gs.headlines.txt headlinesCosine.txt
Pearson: -0.00504
(base) adorinklucas@Adorins-Air sts2016-english-with-gs-v1.0 %  perl correlation-noconfidence.pl STS2016.gs.plagiarism.txt plagiarismCosine.txt
Pearson: 0.15593
(base) adorinklucas@Adorins-Air sts2016-english-with-gs-v1.0 %  perl correlation-noconfidence.pl STS2016.gs.postedit.txt posteditCosine.txt
Pearson: N.A.
(base) adorinklucas@Adorins-Air sts2016-english-with-gs-v1.0 %  perl correlation-noconfidence.pl STS2016.gs.question-question.txt questionCosine.txt
Pearson: 0.12450
(base) adorinklucas@Adorins-Air sts2016-english-with-gs-v1.0 % 
```

For Fine-tuning and training, we have used the **2013**: Test (1,500) Data set.

NOTE: Please find the code for fine tuning attached in NLP_Assignment2_Task2 in Task 2 - Code and Results folder.


C:\WINDOWS\system32\cmd.exe - powershell -Command "& $([scriptblock]::Create((New-Object Net.WebClient).DownloadString('https://platform.activestate.com/dl/cli/_pdli01/install.ps1'))) -c'state activate --...
3°C
Partly cloudy
Search
ENG
IN
19:12
30-11-2022

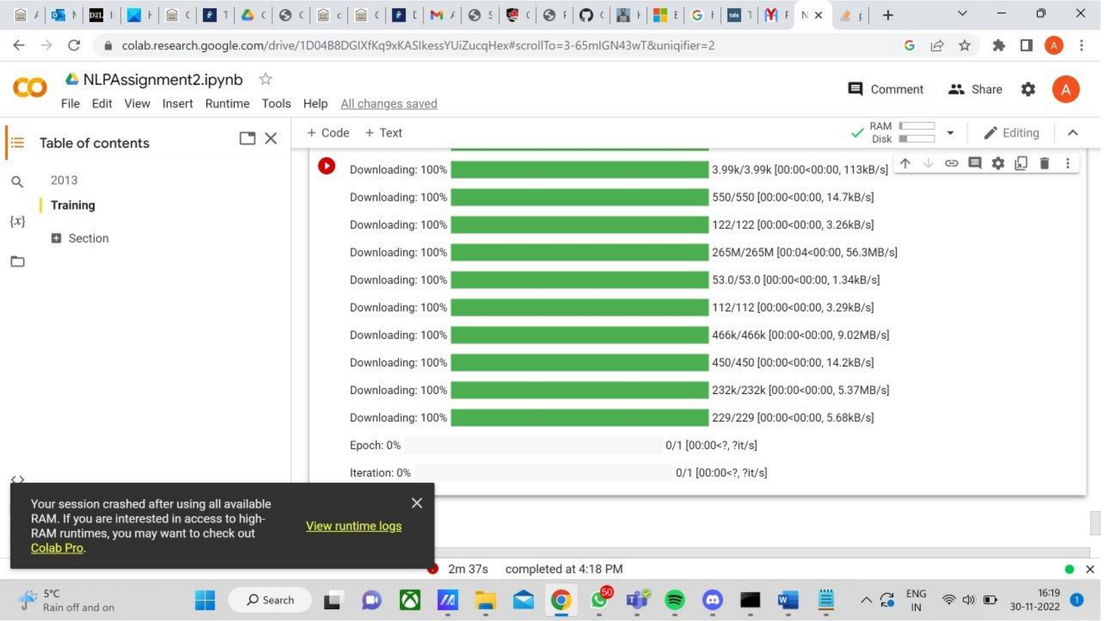
```
[AishwaryaManapuram/Perl-5.36.0-Windows] C:\Users\aishw\Desktop\NLP\Assignment1\New folder>perl correlation-noconfidence.pl STS2016.gs.answer-answer.txt SBERT-Trained-a
nswerCosine.txt
Pearson: 0.19957

[AishwaryaManapuram/Perl-5.36.0-Windows] C:\Users\aishw\Desktop\NLP\Assignment1\New folder>perl correlation-noconfidence.pl STS2016.gs.hperl correlation-noconfidence.pl
 STS2016.gs.headlines.txt SBperl correlation-noconfidence.pl STS2016.gs.headlines.txt SBERT-Trained-heperl correlation-noconfidence.pl STS2016.gs.headlines.txt SBERT-Tr
ained-headlinesCosine.txt
Pearson: 0.01801

[AishwaryaManapuram/Perl-5.36.0-Windows] C:\Users\aishw\Desktop\NLP\Assignment1\New folder>perl correlation-noconfidence.pl STS2016.gs.plagiarism.txt SBERT-Trained-plag
iarismCosine.txt
Pearson: 0.09041

[AishwaryaManapuram/Perl-5.36.0-Windows] C:\Users\aishw\Desktop\NLP\Assignment1\New folder>perl correlation-noconfidence.pl STS2016.gs.postediting.txt SBERT-Trained-pos
teditingCosine.txt
Pearson: 0.02075

[AishwaryaManapuram/Perl-5.36.0-Windows] C:\Users\aishw\Desktop\NLP\Assignment1\New folder>
[AishwaryaManapuram/Perl-5.36.0-Windows] C:\Users\aishw\Desktop\NLP\Assignment1\New folder>perl correlation-noconfidence.pl STS2016.gs.question-question.txt SBERT-Train
ed-questionCosine.txt
Pearson: 0.08436

[AishwaryaManapuram/Perl-5.36.0-Windows] C:\Users\aishw\Desktop\NLP\Assignment1\New folder>
```

Sentence Similarity Results:

Datasets	**SBERT**	**Trained SBERT**
STS2016.input.answer-answer.txt	0.10546	0.19957
STS2016.input.headlines.txt	-0.00504	0.01801
STS2016.input.plagiarism.txt	0.15593	0.09041
STS2016.input.postediting.txt	NA	0.02075
STS2016.input.question-question.txt	0.12450	0.08436

Challenges and Issues faced:

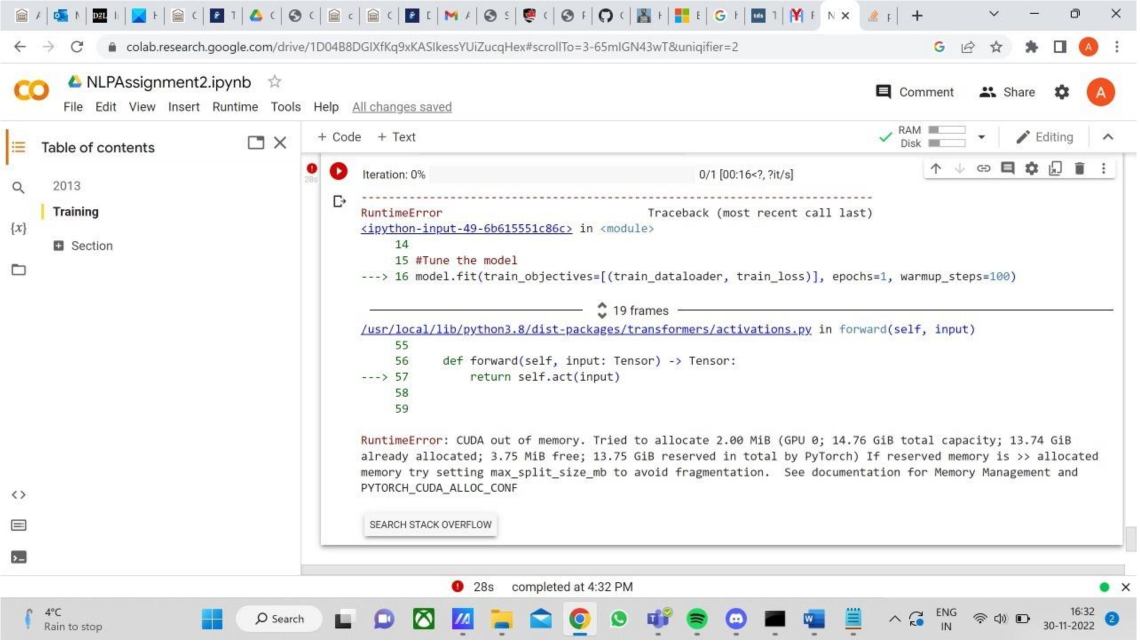
1. We were unable to train the data with multiple data sets due to huge load system crash. We then tried to take 2012 and 2013 datasets also which again resulted in errors due to RAM shortage issues. Hence, we proceeded with 2013 dataset.


colab.research.google.com/drive/1D04B8DGIXfKq9xKASIkessYUiZucqHex#scrollTo=3-65mIGN43wT&uniqifier=2
NLPAssignment2.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
Table of contents
2013
Training
Section
+ Code + Text
RAM Disk
Editing

```
Downloading: 100%  3.99k/3.99k [00:00<00:00, 113kB/s]
Downloading: 100%  550/550 [00:00<00:00, 14.7kB/s]
Downloading: 100%  122/122 [00:00<00:00, 3.26kB/s]
Downloading: 100%  265M/265M [00:04<00:00, 56.3MB/s]
Downloading: 100%  53.0/53.0 [00:00<00:00, 1.34kB/s]
Downloading: 100%  112/112 [00:00<00:00, 3.29kB/s]
Downloading: 100%  466k/466k [00:00<00:00, 9.02MB/s]
Downloading: 100%  450/450 [00:00<00:00, 14.2kB/s]
Downloading: 100%  232k/232k [00:00<00:00, 5.37MB/s]
Downloading: 100%  229/229 [00:00<00:00, 5.68kB/s]
Epoch: 0%  0/1 [00:00<?, ?it/s]
Iteration: 0%  0/1 [00:00<?, ?it/s]
```

Your session crashed after using all available RAM. If you are interested in access to high-RAM runtimes, you may want to check out Colab Pro.

View runtime logs
2m 37s completed at 4:18 PM
5°C Rain off and on
Search
ENG IN
16:19 30-11-2022


colab.research.google.com/drive/1D04B8DGIXfKq9xKASIkessYUiZucqHex#scrollTo=3-65mIGN43wT&uniqifier=2
NLPAssignment2.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
Table of contents
2013
Training
Section
+ Code + Text
RAM Disk
Editing

```
Iteration: 0%  0/1 [00:16<?, ?it/s]

---------------------------------------------------------------------------
RuntimeError                              Traceback (most recent call last)
<ipython-input-49-6b615551c86c> in <module>
     14
     15 #Tune the model
---> 16 model.fit(train_objectives=[(train_dataloader, train_loss)], epochs=1, warmup_steps=100)

                                   19 frames
/usr/local/lib/python3.8/dist-packages/transformers/activations.py in forward(self, input)
     55
     56     def forward(self, input: Tensor) -> Tensor:
---> 57         return self.act(input)
     58
     59

RuntimeError: CUDA out of memory. Tried to allocate 2.00 MiB (GPU 0; 14.76 GiB total capacity; 13.74 GiB
already allocated; 3.75 MiB free; 13.75 GiB reserved in total by PyTorch) If reserved memory is >> allocated
memory try setting max_split_size_mb to avoid fragmentation.  See documentation for Memory Management and
PYTORCH_CUDA_ALLOC_CONF
```

2. We got Pearson correlation values for 2 files (STS2016.input.plagiarism.txt, STS2016.input.question-question.txt) less than the pre-trained model.