README for the DeBERTa fine-tuning code

1. Requirements

- CUDA 10.0
- pytorch 1.3.0
- python 3.6
- bash shell 4.0
- docker
- nvidia-docker2
- 8x32G V100 GPU
- The code has been tested on Nvidia DGX-2 node with Ubuntu 18.04LTS

2. Setup

There are two ways to setup the running environment,

1. Unzip the code to folder DeBERTa, run **pip3 install -r requirements.txt**.
   Then you can try the bash commands under **applications/glue/** folder, e.g. mnli_base.sh
2. Unzip the code to folder DeBERTa, enter the folder, run ./**run_docker.sh to enter docker environment, and then run commands under /DeBERTa/applications/glue/ folder**

Note: To run the commands **you need to enter the folder of the commands first**. If you don't have 8x32G V100 GPU machine, you may need to increase the value of **--accumulative_update 1 in the command file from 1 to 4 or more to apply gradient accumulation to save memory.**

3. Experiments

The results may vary due to different random seeds, whether using FP16, GPU models and drivers. In the table we report median results with different random seeds.

Task	**Command**	**Results**	**Running Time(8x32G V100 GPUs)**
MNLI base	mnli_base.sh	88.8/88.5	1.5h
MNLI large	mnli_large.sh	91.2/91.0	2.5h
QQP large	qqp_large.sh	92.3	6h
QNLI large	qnli_large.sh	95.3	2h

MRPC large	mrpc_large.sh	93.4	0.5h
RTE large	rte_large.sh	88.1	0.5h
SST-2 large	sst2_large.sh	96.7	1h
STS-b large	Stsb_large.sh	92.5	0.5h
CoLA large	cola_large.sh	71.2	0.5h