HOMEWORK 10

Alexandre Olive Pellicer

3.1. BERT for Q&A

Show the train output for first 5 epochs

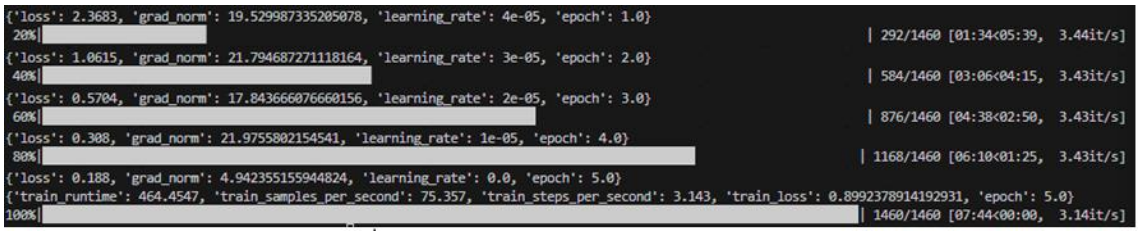

Fig 1: Output after finetuning for 5 epochs

From the obtained output we can see how in each epoch the loss and the learning rate go down. We can also see that the train_loss has been of 0.899 and that the training has last for 7 minutes and 44 seconds.

How are the outputs? Qualitatively look at 10-20 answers and express in your own words how bad or relevant they are. You may need to run more epochs if the sentences make no sense.

These are 15 examples obtained when testing the performance after finetuning for 5 epochs:

```
Question 1: About how many people speak Patois French in St. Barts?
Answer :  500-700
Correct Answer :  500-700
Exact Match :  1
F1 Score :  1.0
---
Question 2: How many people are estimated to have died as a result of the
creation of the Congo-Ocean Railroad?
Answer :  14,000
Correct Answer :  14,000
Exact Match :  1
F1 Score :  1.0
---
Question 3: Where did Chopin prefer to play for people?
Answer :  hisownparisapartmentforsmallgroupsoffriends
Correct Answer :  apartment
Exact Match :  0
F1 Score :  0
---
Question 4: Which contestant came in second on season 12 of American Idol?
```

```
Answer :  candi##ceglover
Correct Answer :  Kree Harrison
Exact Match :  0
F1 Score :  0
---
Question 5: Because of the earthquake, how many people did not have housing?
Answer :  5million
Correct Answer :  at least 5 million
Exact Match :  0
F1 Score :  0
---
Question 6: In what year did Chopin become a French citizen?
Answer :  1835
Correct Answer :  1835
Exact Match :  1
F1 Score :  1.0
---
Question 7: What 007 movie did Sam Mendes previously direct?
Answer :  spec##tre
Correct Answer :  Skyfall
Exact Match :  0
F1 Score :  0
---
Question 8: Who was Ryan Seacrests co-host in the first season of American
Idol?
Answer :  briandun##kle##man
Correct Answer :  Brian Dunkleman
Exact Match :  0
F1 Score :  0
---
Question 9: DNA transposons do not use which genetic material used by Class
1 TEs?
Answer :  rna
Correct Answer :  RNA
Exact Match :  0
F1 Score :  0
---
Question 10: Why did Schumann say the Poles were in mourning?
Answer :
Correct Answer :  the failure of the November 1830
Exact Match :  0
F1 Score :  0
---
Question 11: How many buddhists are in Asia?
Answer :  48##7million
Correct Answer :  487 million
Exact Match :  0
F1 Score :  0
---
Question 12: What organization was Broca in the process of disentangling
himself from?
Answer :  societed'ant##hr##op##olo##giedeparis
Correct Answer :  Société de biologie
Exact Match :  0
F1 Score :  0
---
```

```
Question 13: What kind of service is Tidal?
Answer                                                                  :
musicstreamingservicetidal.theservicespecial##isesinloss##lessaudioandhig
hdefinitionmusicvideos
Correct Answer :  music streaming service
Exact Match :  0
F1 Score :  0
---
Question 14: Who co-wrote Principia Mathematica with Whitehead?
Answer :  bertrandrussell
Correct Answer :  Bertrand Russell
Exact Match :  0
F1 Score :  0
---
Question 15: Who was the winner of American Idols twelfth season?
Answer :  candi##ceglover
Correct Answer :  Candice Glover
Exact Match :  0
F1 Score :  0
```

From the obtained results we can get the following conclusions:

- We can see that when the answer of the question is short and typically a numerical value, the answer of the network tends to match with the correct answer. This is what happens in questions 1, 2, 6 and 9 from the examples presented above.
- We can see that sometimes, normally when the answer is not a numerical value, the answer of the network is close to the correct answer. Sometimes, the output of the network is the correct answer, but words haven't been separated by spaces. This is what happens in questions 13 and 14. Sometimes, the output of the network contains the character "#" in the middle of the answer. This is what happens in questions 8, 11, 12 and 15. Another special case is what happens in question 5. In this case, the answer of the network conceptually matches the correct answer, but it is not written in the same way.
- For questions 3, 4, 7, 9 and 10 we see that the answers of the network are far from the correct answers.

Considering the qualitative evaluation done, we can say that the network shows some evidence that we are in the correct direction but it still needs improvement in order to obtain results that could be used in real world.

3.2. Evaluation Metrics

Calculate the average and median EM and F1-score and report them. For this, you may first calculate individual output in the test set and collect them in a list.

In order to compute the Exact Match Score and the F1 Score we follow the instructions. We first compute the Exact Match Score and the F1 Score for each question, we store the results in two different arrays and, at the end, we compute the average and the median. You can find the code at the end of the report.

```
Average Exact Match: 0.112
Median Exact Match: 0.0
Average F1 Score: 0.12460526315789476
Median F1 Score: 0.0
```

Fig 2: Evaluation metrics after finetuning for 5 epochs

From the obtained results we can see that more than half of the questions have received and answer from the network that has been evaluated with a F1 Score of 0 and an Exact Match Score of 0. Looking at the average results we see that they are close to 0. In general, and considering also the qualitative evaluation done before, we can say that the network shows some evidence that we are in the correct direction but it still needs improvement in order to obtain results that could be used in real world.

Taking into account the obtained results after finetuning the model for 5 epochs, we have considered to increase the number of epochs to see if the performance improved. Thus, we trained the model for 10 epochs. These are the quantitative metrics:

```
Average Exact Match: 0.114
Median Exact Match: 0.0
Average F1 Score: 0.12749415204678363
Median F1 Score: 0.0
```

Fig 3: Evaluation metrics after finetuning for 10 epochs

Although we can see an improvement in the average of the Exact Match Score and the average of the F1 Score, we see that the obtained results are still low. Therefore, we can conclude that the bad performance obtained after finetuning for 5 epochs is not because finetuning for a low amount of epochs but probably because of the model.

In order to get some more insights from the quality of the answers provided by the network, we have also done some post-processing. Considering that in many cases the metrics were showing a bad performance just because the answers of the network contained “#” in the middle of some words, or some words weren’t separated by spaces, or the capital letters didn’t match with the correct answer, we decided to remove these features and compute the metrics again. We did it using the following methods from the String class:

```
.replace("#", "")
.replace(" ", "")
    .lower()
```

Fig 4: Methods used to post process the outputs of the network

Then, when computing the metrics again, these are the results we obtain:

```
Average Exact Match: 0.558
Median Exact Match: 1.0
Average F1 Score: 0.558
Median F1 Score: 1.0
```

Fig 5: Evaluation metrics after setting all letters to low case, removing the “#” and the white spaces

We can see that the metrics are higher. This makes us think that the quality of the predictions is not that bad. We can also see that in this case, since we have removed white spaces from the answers of the network and the correct answers, Exact Match Score and F1 Score will be the same.

3.3. Comparison

Compute and report average and median EM and F1 scores.

```
Average Exact Match: 0.769
Median Exact Match: 1.0
Average F1 Score: 0.890678830586425
Median F1 Score: 1.0
```

Fig 6: Evaluation metrics for the "distilbert-base-cased-distilled-squad" model

The obtained results are much better than the ones obtained when using the "bert-base-uncased" model. It must be said that these results have been computed after not applying any kind of post-processing. In this case we see that more than half of the answers from the network have received F1 Score of 1 and an Exact Match Score of 1. Looking at the average values we see that both Exact Match Score and F1 Score are close to 1. Definitely, these results are better.

We also do a qualitative evaluation of the obtained results. These are 15 examples:

```
Question 1:  From what airport did the chartered flight leave?
Answer :  Taiwan Taoyuan International Airport
Correct Answer :  Taiwan Taoyuan International Airport
Exact Match :  1
F1 Score :  1.0
---
Question 2:  In what historical era does the book take place?
Answer :  the Great Depression
Correct Answer :  the Great Depression
Exact Match :  1
F1 Score :  1.0
---
Question 3:  What is the name of the securities that enabled financial
institutions to obtain investor funds to finance subprime?
Answer :  collateralized debt obligation
Correct Answer :  collateralized debt obligation
Exact Match :  1
F1 Score :  1.0
---
Question 4:  Who tried to spread their territory into Tibet?
Answer :  Dzungar Mongols
Correct Answer :  the Dzungar Mongols
Exact Match :  0
F1 Score :  0.8
---
Question 5:  To whom did Chopin reveal in letters which parts of his work
were about the singing student he was infatuated with?
Answer :  Woyciechowski
Correct Answer :  Tytus Woyciechowski
Exact Match :  0
```

```
F1 Score :  0.6666666666666666
---
Question 6:  How many helicopter were to be provided by the civil
aviation industry?
Answer :  30
Correct Answer :  30
Exact Match :  1
F1 Score :  1.0
---
Question 7:  How many new infections of resistant TB are reported per
year?
Answer :  nearly half a million
Correct Answer :  half a million
Exact Match :  0
F1 Score :  0.8571428571428571
---
Question 8:  What has the Polish government not allowed to find true
cause of death?
Answer :  DNA testing
Correct Answer :  DNA testing
Exact Match :  1
F1 Score :  1.0
---
Question 9:  Which year did PETA spark controversy with Beyonce?
Answer :  2006
Correct Answer :  2006
Exact Match :  1
F1 Score :  1.0
---
Question 10:  What pragmatists did Whitehead acknowledge in the preface
to "Process and Reality"?
Answer :  William James and John Dewey
Correct Answer :  William James and John Dewey
Exact Match :  1
F1 Score :  1.0
---
Question 11:  What three composers did Chopin take inspiration from?
Answer :  J. S. Bach, Mozart and Schubert
Correct Answer :  J. S. Bach, Mozart and Schubert
Exact Match :  1
F1 Score :  1.0
---
Question 12:  What years did the war last through?
Answer :  1992 to 1997
Correct Answer :  1992 to 1997
Exact Match :  1
F1 Score :  1.0
---
Question 13:  Who designed Chopin's tombstone?
Answer :  Clésinger
Correct Answer :  Clésinger.
Exact Match :  0
F1 Score :  0
---
Question 14:  What was Beyonce's 2010  perfume called?
Answer :  Heat
```

```
Correct Answer :  Heat
Exact Match :  1
F1 Score :  1.0
---
Question 15:  The Brooklyn Bridge was the worlds largest until what date?
Answer :  1903
Correct Answer :  1903
Exact Match :  1
F1 Score :  1.0
---
```

From the obtained results we can see that the answers from the network very often match with the correct answers. In some cases where the F1 Score is not 1 is because the network has added some extra words in the answer. Nevertheless, the added words still makes sense so that from a qualitative perspective, the answer could also be considered as correct.

CODE

```
from transformers import BertForQuestionAnswering
from transformers import TrainingArguments
import pickle
from transformers import Trainer
from datasets import Dataset
import pandas as pd
import numpy as np
from transformers import BertTokenizer
from transformers import pipeline

model_name = 'bert-base-uncased'
model = BertForQuestionAnswering.from_pretrained(model_name)

training_args = TrainingArguments (
    output_dir ='./results', # output directory
    use_mps_device = False ,
    num_train_epochs =5 , # total number of training epochs , change this
as you need
    per_device_train_batch_size=8 , # batch size per device during
training ,change this as you need
    per_device_eval_batch_size=8 , # batch size for evaluation , change
this as you need
    weight_decay =0.01 , # strength of weight decay
    logging_dir ='./logs', # directory for storing logs
    logging_strategy="epoch",
    save_strategy="epoch"
)

with open('/home/aolivepe/ECE60146/HW10/dataset/train_dict.pkl','rb') as
f:
    train_dict = pickle.load(f)
```

```
with open('/home/aolivepe/ECE60146/HW10/dataset/test_dict.pkl','rb') as
f:
    test_dict = pickle.load(f)
with open('/home/aolivepe/ECE60146/HW10/dataset/eval_dict.pkl','rb') as
f:
    eval_dict = pickle.load(f)
with
open('/home/aolivepe/ECE60146/HW10/dataset/train_data_processed.pkl','rb'
) as f:
    train_processed = pickle.load(f)
with
open('/home/aolivepe/ECE60146/HW10/dataset/test_data_processed.pkl','rb')
as f:
    test_processed = pickle.load(f)
with
open('/home/aolivepe/ECE60146/HW10/dataset/eval_data_processed.pkl','rb')
as f:
    eval_processed = pickle.load(f)

train_dataset = Dataset.from_pandas(pd.DataFrame(train_processed))
eval_dataset = Dataset.from_pandas(pd.DataFrame(eval_processed))
test_dataset = Dataset.from_pandas(pd.DataFrame(test_processed))
trainer = Trainer(
    model=model , # the instantiated Transformers model to be fine -
tuned
    args = training_args , # training arguments , defined above
    train_dataset = train_dataset , # training dataset
    eval_dataset = eval_dataset # evaluation dataset
)

trainer.train()

## QUALITATIVE AND QUANTITATIVE EVALUATION ----------------------------
----------------------------------------
def compute_exact_match(prediction, truth):
    return int(prediction == truth)

def f1_score(prediction, truth):
    pred_tokens = prediction.split()
    truth_tokens = truth.split()
    # if either the prediction or the truth is no - answer then f1 = 1 if
they agree , 0 otherwise
    if len(pred_tokens) == 0 or len(truth_tokens) == 0:
        return int(pred_tokens == truth_tokens)
    common_tokens = set(pred_tokens) & set(truth_tokens)
    # if there are no common tokens then f1 = 0
    if len(common_tokens) == 0:
        return 0
    prec = len(common_tokens)/len(pred_tokens)
```

```
    rec = len(common_tokens)/len(truth_tokens)
    return 2*(prec*rec)/(prec+rec)

x = trainer.predict(test_dataset)
start_pos, end_pos = x.predictions
start_pos = np.argmax(start_pos, axis =1)
end_pos = np.argmax(end_pos, axis =1)
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')

# Arrays where we store Exact Match Score and F1 Score for each question
em = []
f1 = []

for k, (i, j) in enumerate(zip(start_pos, end_pos)):
    tokens =
tokenizer.convert_ids_to_tokens(test_processed['input_ids'][k])
    print('Question :', test_dict['question'][k])
    print('Answer : ',''.join(tokens[i:j+1]))
    print('Correct Answer : ', test_dict['answers'][k]['text'][0])
    print('Exact Match : ', compute_exact_match(''.join(tokens[i:j+1]),
test_dict['answers'][k]['text'][0]))
    print('F1 Score : ', f1_score(''.join(tokens[i:j+1]),
test_dict['answers'][k]['text'][0]))
    print('---')
    em.append(compute_exact_match(''.join(tokens[i:j+1]),
test_dict['answers'][k]['text'][0]))
    f1.append(f1_score(''.join(tokens[i:j+1]),
test_dict['answers'][k]['text'][0]))

def compute_average(arr):
    total = sum(arr)
    return total / len(arr)

def compute_median(arr):
    sorted_arr = sorted(arr)
    n = len(arr)
    if n % 2 == 0:
        median = (sorted_arr[n // 2 - 1] + sorted_arr[n // 2]) / 2
    else:
        median = sorted_arr[n // 2]
    return median

# Compute average
average_em = compute_average(em)
print("Average Exact Match:", average_em)

# Compute median
median_em = compute_median(em)
print("Median Exact Match:", median_em)
```

```
# Compute average
average_f1 = compute_average(f1)
print("Average F1 Score:", average_f1)

# Compute median
median_f1 = compute_median(f1)
print("Median F1 Score:", median_f1)

## COMPARISON ------------------------------------------------------------
---------------------------------
question_answerer = pipeline("question-answering", model='distilbert-
base-cased-distilled-squad')

# Arrays where we store Exact Match Score and F1 Score for each question
em = []
f1 = []

for i in range(len(test_dict['question'])):
    result = question_answerer(question = test_dict['question'][i],
context = test_dict['context'][i])
    print('Question : ', test_dict['question'][i])
    print('Answer : ', result['answer'])
    print('Correct Answer : ', test_dict['answers'][i]['text'][0])
    print('Exact Match : ',
compute_exact_match(result['answer'],test_dict['answers'][i]['text'][0]))
    print('F1 Score : ', f1_score(result['answer'],
test_dict['answers'][i]['text'][0]))
    print('---')
    em.append(compute_exact_match(result['answer'],test_dict['answers'][i
]['text'][0]))
    f1.append(f1_score(result['answer'],
test_dict['answers'][i]['text'][0]))

# Compute average
average_em = compute_average(em)
print("Average Exact Match:", average_em)

# Compute median
median_em = compute_median(em)
print("Median Exact Match:", median_em)

# Compute average
average_f1 = compute_average(f1)
print("Average F1 Score:", average_f1)

# Compute median
median_f1 = compute_median(f1)
print("Median F1 Score:", median_f1)
```