

▼ BERTScore

This notebook demonstrates the evaluation of a model on test data during training.

```
!pip install transformers==3.0.1 bert_score==0.3.9
from bert_score import score
```

```

Collecting transformers==3.0.1
  Downloading https://files.pythonhosted.org/packages/82/25/89050e69ed53c2a3b
    |████████████████████████████████████████| 757kB 35.2MB/s
Collecting bert_score==0.3.9
  Downloading https://files.pythonhosted.org/packages/38/fb/e63e7e231a79db048
    |████████████████████████████████████████| 61kB 10.3MB/s
Requirement already satisfied: packaging in /usr/local/lib/python3.7/dist-pac
Collecting sacremoses
  Downloading https://files.pythonhosted.org/packages/75/ee/67241dc87f266093c
    |████████████████████████████████████████| 901kB 48.6MB/s
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: filelock in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-pack
Collecting tokenizers==0.8.0-rc4
  Downloading https://files.pythonhosted.org/packages/f7/82/0e82a95bd9db2b325
    |████████████████████████████████████████| 3.0MB 48.0MB/s
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.7/
Collecting sentencepiece!=0.1.92
  Downloading https://files.pythonhosted.org/packages/ac/aa/1437691b0c7c83086
    |████████████████████████████████████████| 1.2MB 37.0MB/s
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: torch>=1.0.0 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: pandas>=1.0.1 in /usr/local/lib/python3.7/dist
Requirement already satisfied: pyparsing>=2.0.2 in /usr/local/lib/python3.7/d
Requirement already satisfied: joblib in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: click in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in /us
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3
Requirement already satisfied: cycloper>=0.10 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.7/
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/dist-
Installing collected packages: sacremoses, tokenizers, sentencepiece, transfo
Successfully installed bert-score-0.3.9 sacremoses-0.0.45 sentencepiece-0.1.9

```

BERTScore on Week 8's Assignment

▼ Installing Modules

```
!pip install pytorch_lightning torchmetrics tableprint spacy==3
!python -m spacy download en_core_web_sm
!python -m spacy download de_core_news_sm
```

```

Collecting pytorch_lightning
  Downloading https://files.pythonhosted.org/packages/48/5e/19c817ad2670c1d82
|████████████████████████████████████████| 819kB 35.0MB/s
Collecting torchmetrics
  Downloading https://files.pythonhosted.org/packages/32/44/e21e1bc2f0a2955ab
|████████████████████████████████████████| 235kB 34.6MB/s
Collecting tableprint
  Downloading https://files.pythonhosted.org/packages/e4/9f/e61234cc4a1e2c7c8
Collecting spacy==3
  Downloading https://files.pythonhosted.org/packages/8b/62/a98c61912ea573448
|████████████████████████████████████████| 12.7MB 240kB/s
Collecting pyDeprecate==0.3.0
  Downloading https://files.pythonhosted.org/packages/14/52/aa227a0884df71ed1
Requirement already satisfied: numpy>=1.17.2 in /usr/local/lib/python3.7/dist
Collecting future>=0.17.1
  Downloading https://files.pythonhosted.org/packages/45/0b/38b06fd9b92dc2b68
|████████████████████████████████████████| 829kB 46.9MB/s
Collecting fsspec[http]!=2021.06.0,>=2021.05.0
  Downloading https://files.pythonhosted.org/packages/0e/3a/666e63625a19883ae
|████████████████████████████████████████| 122kB 57.6MB/s
Collecting PyYAML<=5.4.1,>=5.1
  Downloading https://files.pythonhosted.org/packages/7a/a5/393c087efdc78091a
|████████████████████████████████████████| 645kB 53.3MB/s
Requirement already satisfied: torch>=1.4 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: packaging>=17.0 in /usr/local/lib/python3.7/di
Requirement already satisfied: pillow!=8.3.0 in /usr/local/lib/python3.7/dist
Collecting tensorboard!=2.5.0,>=2.2.0
  Downloading https://files.pythonhosted.org/packages/64/21/eebd23060763fedee
|████████████████████████████████████████| 10.6MB 211kB/s
Requirement already satisfied: tqdm>=4.41.0 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: wcwidth in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.
Collecting catalogue<2.1.0,>=2.0.1
  Downloading https://files.pythonhosted.org/packages/9c/10/dbc1203a4b1367c7b
Collecting thinc<8.1.0,>=8.0.0
  Downloading https://files.pythonhosted.org/packages/7a/6e/bd2da3d71ab2d1752
|████████████████████████████████████████| 624kB 58.7MB/s
Requirement already satisfied: blis<0.8.0,>=0.4.0 in /usr/local/lib/python3.7
Collecting pydantic<1.8.0,>=1.7.1
  Downloading https://files.pythonhosted.org/packages/ca/fa/d43f31874e1f2a963
|████████████████████████████████████████| 9.1MB 50.7MB/s
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/py
Requirement already satisfied: jinja2 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: typing-extensions>=3.7.4; python_version < "3.
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/pyth
Collecting pathy
  Downloading https://files.pythonhosted.org/packages/65/ae/ecfa3e2dc267010fa
|████████████████████████████████████████| 51kB 8.3MB/s
Collecting spacy-legacy<3.1.0,>=3.0.0
  Downloading https://files.pythonhosted.org/packages/fa/a5/a727792d000b2a7bf
Requirement already satisfied: setuptools in /usr/local/lib/python3.7/dist-pa
Collecting srsly<3.0.0,>=2.4.0
  Downloading https://files.pythonhosted.org/packages/c3/84/dfdfc9f6f04f6b882
|████████████████████████████████████████| 460kB 53.1MB/s
Requirement already satisfied: importlib-metadata>=0.20; python_version < "3.
Collecting typer<0.4.0,>=0.3.0
  Downloading https://files.pythonhosted.org/packages/90/34/d138832f6945432c6
Requirement already satisfied: wasabi<1.1.0,>=0.8.1 in /usr/local/lib/python3
Collecting aiohttp; extra == "http"
  Downloading https://files.pythonhosted.org/packages/88/c0/5890b4c8b04a79b73

```

```

|████████████████████████████████████████| 1.3MB 50.7MB/s
Requirement already satisfied: pyparsing>=2.0.2 in /usr/local/lib/python3.7/d
Requirement already satisfied: werkzeug>=0.11.15 in /usr/local/lib/python3.7/
Requirement already satisfied: six>=1.10.0 in /usr/local/lib/python3.7/dist-p
Requirement already satisfied: protobuf>=3.6.0 in /usr/local/lib/python3.7/di
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in /usr/local/li
Requirement already satisfied: google-auth<2,>=1.6.3 in /usr/local/lib/python
Requirement already satisfied: wheel>=0.26; python_version >= "3" in /usr/loc
Requirement already satisfied: absl-py>=0.4 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in /usr/local
Requirement already satisfied: grpcio>=1.24.3 in /usr/local/lib/python3.7/dis
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.7/di
Requirement already satisfied: zipp>=0.5; python_version < "3.8" in /usr/loca
Requirement already satisfied: MarkupSafe>=0.23 in /usr/local/lib/python3.7/d
Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in /us
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/
Requirement already satisfied: smart-open<6.0.0,>=5.0.0 in /usr/local/lib/pyt
Requirement already satisfied: click<7.2.0,>=7.1.1 in /usr/local/lib/python3.
Collecting multidict<7.0,>=4.5
  Downloading https://files.pythonhosted.org/packages/7c/a6/4123b8165acbe773d
|████████████████████████████████████████| 143kB 58.1MB/s
Collecting yarll<2.0,>=1.0
  Downloading https://files.pythonhosted.org/packages/f1/62/046834c5fc998c88a
|████████████████████████████████████████| 296kB 57.6MB/s
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.7/dist
Collecting async-timeout<4.0,>=3.0
  Downloading https://files.pythonhosted.org/packages/e1/1e/5a4441be21b0726c4
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python
Requirement already satisfied: cachetools<5.0,>=2.0.0 in /usr/local/lib/pytho
Requirement already satisfied: rsa<5,>=3.1.4; python_version >= "3.6" in /usr
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/pyt
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in /usr/local/lib/python3
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.7/di
Building wheels for collected packages: future
  Building wheel for future (setup.py) ... done
  Created wheel for future: filename=future-0.18.2-cp37-none-any.whl size=491
  Stored in directory: /root/.cache/pip/wheels/8b/99/a0/81daf51dcd359a9377b11
Successfully built future
ERROR: tensorflow 2.5.0 has requirement tensorboard~2.5, but you'll have ten
Installing collected packages: pyDeprecate, future, multidict, yarll, async-ti
  Found existing installation: future 0.16.0
    Uninstalling future-0.16.0:
      Successfully uninstalled future-0.16.0
  Found existing installation: PyYAML 3.13
    Uninstalling PyYAML-3.13:
      Successfully uninstalled PyYAML-3.13
  Found existing installation: tensorboard 2.5.0
    Uninstalling tensorboard-2.5.0:
      Successfully uninstalled tensorboard-2.5.0
  Found existing installation: catalogue 1.0.0
    Uninstalling catalogue-1.0.0:
      Successfully uninstalled catalogue-1.0.0
  Found existing installation: srsly 1.0.5
    Uninstalling srsly-1.0.5:
      Successfully uninstalled srsly-1.0.5
  Found existing installation: thinc 7.4.0
    Uninstalling thinc-7.4.0:
      Successfully uninstalled thinc-7.4.0
  Found existing installation: spacy 2.2.4
    Uninstalling spacy-2.2.4:

```

Successfully uninstalled spacy-2.2.4
 Successfully installed PyYAML-5.4.1 aiohttp-3.7.4.post0 async-timeout-3.0.1 c
WARNING: The following packages were previously imported in this runtime:
[tensorboard,yaml]
You must restart the runtime in order to use newly installed versions.

RESTART RUNTIME

2021-07-05 06:49:03.455261: I tensorflow/stream_executor/platform/default/dso
 Collecting en-core-web-sm==3.0.0

Downloading <https://github.com/explosion/spacy-models/releases/download/en->
 |██| 13.7MB 224kB/s

Requirement already satisfied: spacy<3.1.0,>=3.0.0 in /usr/local/lib/python3.
 Requirement already satisfied: typer<0.4.0,>=0.3.0 in /usr/local/lib/python3.
 Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/py
 Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.
 Requirement already satisfied: blis<0.8.0,>=0.4.0 in /usr/local/lib/python3.7
 Requirement already satisfied: wasabi<1.1.0,>=0.8.1 in /usr/local/lib/python3
 Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/pyth
 Requirement already satisfied: setuptools in /usr/local/lib/python3.7/dist-pa
 Requirement already satisfied: typing-extensions>=3.7.4; python_version < "3.
 Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.0 in /usr/local/lib/p
 Requirement already satisfied: srsly<3.0.0,>=2.4.0 in /usr/local/lib/python3.
 Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /usr/local/lib/python3.
 Requirement already satisfied: importlib-metadata>=0.20; python_version < "3.
 Requirement already satisfied: jinja2 in /usr/local/lib/python3.7/dist-packag
 Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python
 Requirement already satisfied: pydantic<1.8.0,>=1.7.1 in /usr/local/lib/pytho
 Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.7/dist
 Requirement already satisfied: pathy in /usr/local/lib/python3.7/dist-package
 Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.7/di
 Requirement already satisfied: catalogue<2.1.0,>=2.0.1 in /usr/local/lib/pyth
 Requirement already satisfied: thinc<8.1.0,>=8.0.0 in /usr/local/lib/python3.
 Requirement already satisfied: click<7.2.0,>=7.1.1 in /usr/local/lib/python3.
 Requirement already satisfied: certifi=2017.4.17 in /usr/local/lib/python3.7

▼ Imports

Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-pac

Import Libraries

```
import random
from typing import Iterable, List, Tuple
import pandas as pd
import sys, os, pickle
import math
import matplotlib.pyplot as plt
import numpy as np
import spacy
```

PyTorch related

```
import torch, torchtext
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
from torch import Tensor
from torchtext.data.utils import get_tokenizer
from torchtext.vocab import build_vocab_from_iterator
from torchtext.datasets import Multi30k
from torch.nn.utils.rnn import pad_sequence
```

```

from torch.utils.data import DataLoader

# My Custom Code
import pytorch_lightning as pl
import torchmetrics
from pytorch_lightning.loggers import CSVLogger
from pytorch_lightning.callbacks import ModelCheckpoint
import tableprint as tp

requirements already satisfied: pandas==1.1.5 in /usr/local/lib/python3.7/dist-packages
# Manual Seed
SEED = 1234

random.seed(SEED)
np.random.seed(SEED)
torch.manual_seed(SEED)
torch.cuda.manual_seed(SEED)
torch.backends.cudnn.deterministic = True
Successfully installed de-core-news-sm-3.0.0

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")

```

▼ Language Definitions

```

SRC_LANGUAGE = 'de'
TGT_LANGUAGE = 'en'

# Place-holders
token_transform = {}
vocab_transform = {}

```

▼ Tokenizers

```

token_transform[SRC_LANGUAGE] = get_tokenizer('spacy', language='de_core_news_sm')
token_transform[TGT_LANGUAGE] = get_tokenizer('spacy', language='en_core_web_sm')

```

▼ Yield Function

This yields the tokens for the texts and will be used to build the vocab

```

def yield_tokens(data_iter: Iterable, language: str) -> List[str]:
    language_index = {SRC_LANGUAGE: 0, TGT_LANGUAGE: 1}

    for data_sample in data_iter:
        yield token_transform[language](data_sample[language_index[language]])

```

▼ Special Tokens

```
# Define special symbols and indices
UNK_IDX, PAD_IDX, BOS_IDX, EOS_IDX = 0, 1, 2, 3
# Make sure the tokens are in order of their indices to properly insert them in voc
special_symbols = ['<unk>', '<pad>', '<bos>', '<eos>']
```

Build the vocab here

```
for ln in [SRC_LANGUAGE, TGT_LANGUAGE]:
    # Training data Iterator
    train_iter = Multi30k(split='train', language_pair=(SRC_LANGUAGE, TGT_LANGUAGE))
    # Create torchtext's Vocab object
    vocab_transform[ln] = build_vocab_from_iterator(yield_tokens(train_iter, ln),
                                                    min_freq=1,
                                                    specials=special_symbols,
                                                    special_first=True)
```

```
training.tar.gz: 100%|██████████| 1.21M/1.21M [00:00<00:00, 1.64MB/s]
```

▼ Setting the default index as the token

```
# Set UNK_IDX as the default index. This index is returned when the token is not f
# If not set, it throws RuntimeError when the queried token is not found in the Voi
for ln in [SRC_LANGUAGE, TGT_LANGUAGE]:
    vocab_transform[ln].set_default_index(UNK_IDX)
```

```
len(vocab_transform['de'])
```

```
19215
```

```
len(vocab_transform['en'])
```

```
10838
```

▼ Collator

```
# helper function to club together sequential operations
def sequential_transforms(*transforms):
    def func(txt_input):
        for transform in transforms:
            txt_input = transform(txt_input)
        return txt_input
    return func
```

```

# function to add BOS/EOS and create tensor for input sequence indices
def tensor_transform(token_ids: List[int]):
    return torch.cat((torch.tensor([BOS_IDX]),
                        torch.tensor(token_ids),
                        torch.tensor([EOS_IDX])))

# src and tgt language text transforms to convert raw strings into tensors indices
text_transform = {}
for ln in [SRC_LANGUAGE, TGT_LANGUAGE]:
    text_transform[ln] = sequential_transforms(token_transform[ln], #Tokenization
                                              vocab_transform[ln], #Numericalization
                                              tensor_transform) # Add BOS/EOS and

# function to collate data samples into batch tensors
def collate_fn(batch):
    src_batch, tgt_batch = [], []
    for src_sample, tgt_sample in batch:
        src_batch.append(text_transform[SRC_LANGUAGE](src_sample.rstrip("\n")))
        tgt_batch.append(text_transform[TGT_LANGUAGE](tgt_sample.rstrip("\n")))

    src_batch = pad_sequence(src_batch, padding_value=PAD_IDX)
    tgt_batch = pad_sequence(tgt_batch, padding_value=PAD_IDX)
    return src_batch, tgt_batch

```

▼ DataLoader

```

BATCH_SIZE = 32
train_iter = Multi30k(split='train', language_pair=(SRC_LANGUAGE, TGT_LANGUAGE))
train_loader = DataLoader(train_iter, batch_size=BATCH_SIZE, collate_fn=collate_fn)

val_iter = Multi30k(split='valid', language_pair=(SRC_LANGUAGE, TGT_LANGUAGE))
val_loader = DataLoader(val_iter, batch_size=BATCH_SIZE, collate_fn=collate_fn, num_workers=4)

test_iter = Multi30k(split='test', language_pair=(SRC_LANGUAGE, TGT_LANGUAGE))
test_loader = DataLoader(test_iter, batch_size=BATCH_SIZE, collate_fn=collate_fn, num_workers=4)

validation.tar.gz: 100%|██████████| 46.3k/46.3k [00:00<00:00, 270kB/s]
mmt16_task1_test.tar.gz: 100%|██████████| 43.9k/43.9k [00:00<00:00, 263kB/s]

```

▼ Model

▼ Boilerplate Code for PyTorch Lightning

```

class TL(pl.LightningModule):
    def __init__(self):
        super(TL, self).__init__()

        self.train_loader = train_loader
        self.val_loader = val_loader
        self.test_loader = test_loader

```



```

self.train_acc = torch.tensor(0.)
self.avg_train_loss = torch.tensor(0.)
self.table_context = None
self.trgs = []
self.preds = []

def training_step(self, batch, batch_idx):
    src, trg = batch
    output = self(src, trg)
    output_dim = output.shape[-1]
    output = output[1:].view(-1, output_dim)
    trg = trg[1:].view(-1)
    loss_train = self.loss(output, trg)
    return loss_train

def validation_step(self, batch, batch_idx):
    src, trg = batch
    output = self(src, trg, 0)

    out = output.argmax(2)

    o = torch.transpose(out,0,1)
    t = torch.transpose(trg,0,1)
    for o1, t1 in zip(o,t):
        stop_ind_trg = (t1==3).nonzero()[0].item() # stop when <eos> token is found
        if any(o1==3) == False: # if <eos> token is not found
            stop_ind_pred = len(o1) # use complete sentence
        else:
            stop_ind_pred = (o1==3).nonzero()[0].item() # stop when <eos> token is found

        trg_sent_i = t1[:stop_ind_trg+1]
        pred_sent_i = o1[:stop_ind_pred+1]

        trg_sent_tok = [vocab_transform['en'].lookup_token(word_i) for word_i in trg_sent_i]
        pred_sent_tok = [vocab_transform['en'].lookup_token(word_i) for word_i in pred_sent_i]

        trg_sent = " ".join(trg_sent_tok)
        pred_sent = " ".join(pred_sent_tok)

        self.trgs.append(trg_sent)
        self.preds.append(pred_sent)

    output_dim = output.shape[-1]
    output = output[1:].view(-1, output_dim)
    trg = trg[1:].view(-1)
    loss_valid = self.loss(output, trg)

    return {"loss": loss_valid}

def training_epoch_end(self, outputs):

```

```

def training_epoch_end(self, outputs):
    self.avg_train_loss = torch.stack([x['loss'] for x in outputs]).mean()

def validation_epoch_end(self, outputs):
    if trainer.sanity_checking:
        print('sanity check')
        return
    P, R, F1 = score(self.preds, self.trgs, lang="en", verbose=False)
    P, R, F1 = round(P.mean().item(), 3), round(R.mean().item(), 3), round(F1.mean().item(), 3)

    self.trgs = []
    self.preds = []

    avg_valid_loss = torch.stack([x['loss'] for x in outputs]).mean()
    metrics = {'epoch': self.current_epoch+1, 'Train PPL': math.exp(self.avg_train_loss.item()), 'Valid PPL': math.exp(avg_valid_loss.item()), 'F1': F1}
    if self.table_context is None:
        self.table_context = tp.TableContext(headers=['epoch', 'Train PPL', 'Valid PPL', 'F1'])
        self.table_context.__enter__()
    self.table_context([self.current_epoch+1, math.exp(self.avg_train_loss.item()), math.exp(avg_valid_loss.item()), F1])
    self.logger.log_metrics(metrics)
    if self.current_epoch == self.trainer.max_epochs - 1:
        self.validation_end(outputs)

def validation_end(self, outputs):
    self.table_context.__exit__()

```

▼ Encoder

```

class Encoder(pl.LightningModule):
    def __init__(self, input_dim, emb_dim, hid_dim, dropout):
        super().__init__()

        self.hid_dim = hid_dim

        self.embedding = nn.Embedding(input_dim, emb_dim)
        self.rnn = nn.GRU(emb_dim, hid_dim)
        self.dropout = nn.Dropout(dropout)

    def forward(self, src):
        embedded = self.dropout(self.embedding(src))
        output, hidden = self.rnn(embedded)

        return hidden

```

▼ Decoder

```

class Decoder(pl.LightningModule):
    def __init__(self, output_dim, emb_dim, hid_dim, dropout):
        super().__init__()

        self.hid_dim = hid_dim

```

```

self.output_dim = output_dim
self.embedding = nn.Embedding(output_dim, emb_dim)
self.rnn = nn.GRU(emb_dim + hid_dim, hid_dim)
self.fc_out = nn.Linear(emb_dim + hid_dim * 2, output_dim)
self.dropout = nn.Dropout(dropout)

def forward(self, input, hidden, context):
    input = input.unsqueeze(0)
    embedded = self.dropout(self.embedding(input))
    emb_con = torch.cat((embedded, context), dim = 2)
    output, hidden = self.rnn(emb_con, hidden)
    output = torch.cat((embedded.squeeze(0), hidden.squeeze(0), context.squeeze(0)), dim = 0)
    prediction = self.fc_out(output)
    return prediction, hidden

```

▼ Seq2Seq Model

Define the model

```

class Seq2Seq(TL):
    def __init__(self, encoder, decoder, device):
        super(Seq2Seq, self).__init__()

        self.loss = nn.CrossEntropyLoss(ignore_index=PAD_IDX)
        self.lr = 1e-3

        self.encoder = encoder
        self.decoder = decoder
        # self.device = device # Doesn't work in PyTorchLightning since it is already on the GPU

        assert encoder.hid_dim == decoder.hid_dim, "Hidden Dimensions of Encoder and Decoder must be the same"

    def forward(self, src, trg, teacher_forcing_ratio = 0.5):

        batch_size = trg.shape[1]
        trg_len = trg.shape[0]
        trg_vocab_size = self.decoder.output_dim

        outputs = torch.zeros(trg_len, batch_size, trg_vocab_size).to(self.device)

        context = self.encoder(src)
        hidden = context

        input = trg[0,:]

        for t in range(1, trg_len):

            output, hidden = self.decoder(input, hidden, context)

            outputs[t] = output

            teacher_force = random.random() < teacher_forcing_ratio

```

```

        top1 = output.argmax(1)

        input = trg[t] if teacher_force else top1

    return outputs

def configure_optimizers(self):
    optim = torch.optim.Adam(self.parameters())
    return optim

```

▼ Model Initialization and Summary

```

INPUT_DIM = len(vocab_transform[SRC_LANGUAGE])
OUTPUT_DIM = len(vocab_transform[TGT_LANGUAGE])

ENC_EMB_DIM = 256
DEC_EMB_DIM = 256
HID_DIM = 512
ENC_DROPOUT = 0.5
DEC_DROPOUT = 0.5

enc = Encoder(INPUT_DIM, ENC_EMB_DIM, HID_DIM, ENC_DROPOUT)
dec = Decoder(OUTPUT_DIM, DEC_EMB_DIM, HID_DIM, DEC_DROPOUT)

model = Seq2Seq(enc, dec, device).to(device)

```

▼ Model Checkpoint

```

checkpoint_callback = ModelCheckpoint(
    monitor='val_loss',
    dirpath='/content',
    filename='sst-{epoch:02d}-{val_loss:.2f}',
    mode='min'
)

!rm -rf csv_logs
csvlogger = CSVLogger('csv_logs', name='END2_Assign_9', version=0)
trainer = pl.Trainer(max_epochs=5, num_sanity_val_steps=0, logger=csvlogger, gpus=1)
trainer.fit(model, train_dataloader=train_loader, val_dataloaders=val_loader)
checkpoint_callback.best_model_path

```

GPU available: True, used: True
 TPU available: False, using: 0 TPU cores
 LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES: [0]

	Name	Type	Params
0	loss	CrossEntropyLoss	0
1	encoder	Encoder	6.1 M
2	decoder	Decoder	18.6 M

24.7 M Trainable params

0 Non-trainable params

24.7 M Total params

98.916 Total estimated model params size (MB)

/usr/local/lib/python3.7/dist-packages/pytorch_lightning/utilities/data.py:42

'Your `IterableDataset` has `__len__` defined.'

Epoch 4: 100% 939/939 [01:49<00:00, 8.57it/s, loss=2.39, v_num=0]

Downloading: 100% 482/482 [00:00<00:00, 5.89kB/s]

Downloading: 100% 899k/899k [00:00<00:00, 3.56MB/s]

Downloading: 100% 456k/456k [00:00<00:00, 4.55MB/s]

Downloading: 100% 1.43G/1.43G [04:05<00:00, 5.81MB/s]

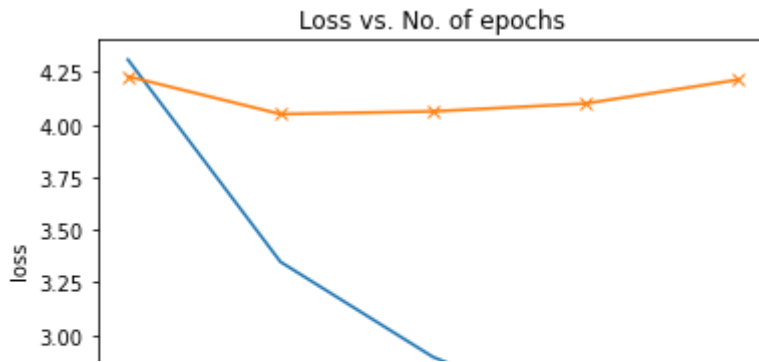
epoch	Train PPL	Train Loss	Valid PPL	Valid Loss	V
1	74.355	4.3088	68.656	4.2291	
2	28.446	3.348	57.394	4.0499	
3	18.098	2.8958	58.138	4.0628	

▼ Training Log

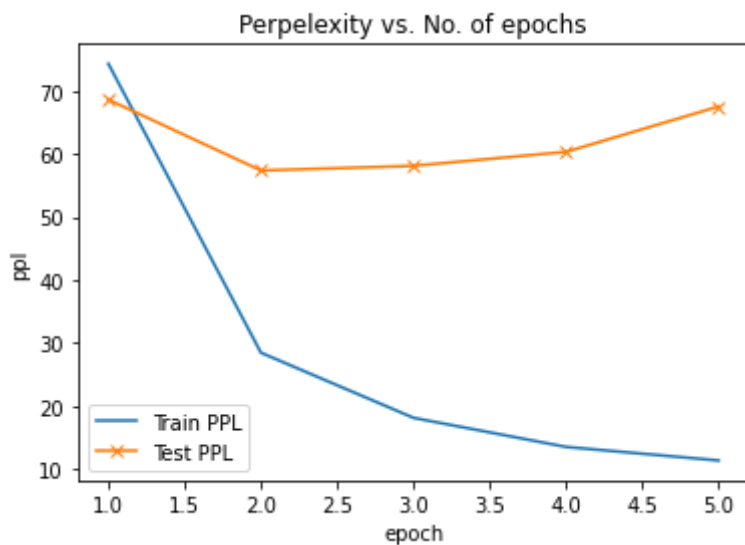
```

..
root='./csv_logs/' + 'END2_Assign_9' + '/'
dirlist = [ item for item in os.listdir(root) if os.path.isdir(os.path.join(root,
metricfile = root + dirlist[-1:][0] + '/metrics.csv'
metrics = pd.read_csv(metricfile)

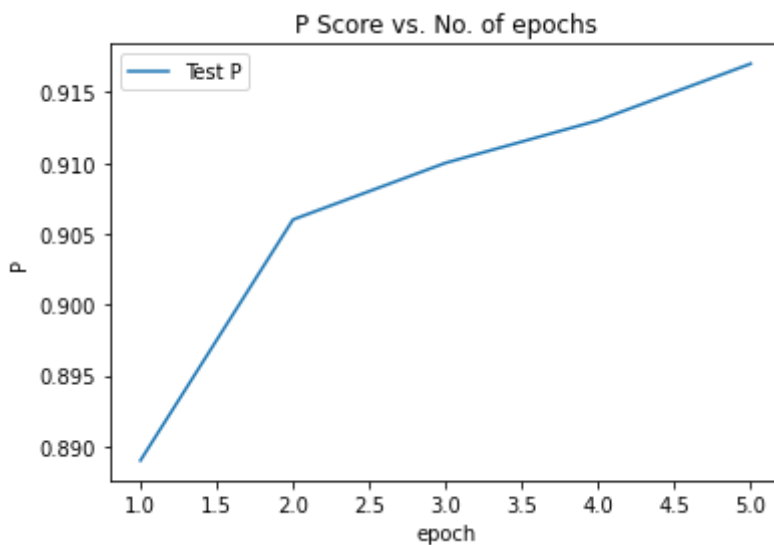
plt.plot(metrics['epoch'], metrics['Train Loss'], label="Train Loss")
plt.plot(metrics['epoch'], metrics['Valid Loss'], '-x', label="Test Loss")
plt.xlabel('epoch')
plt.ylabel('loss')
plt.legend()
plt.title('Loss vs. No. of epochs');
```



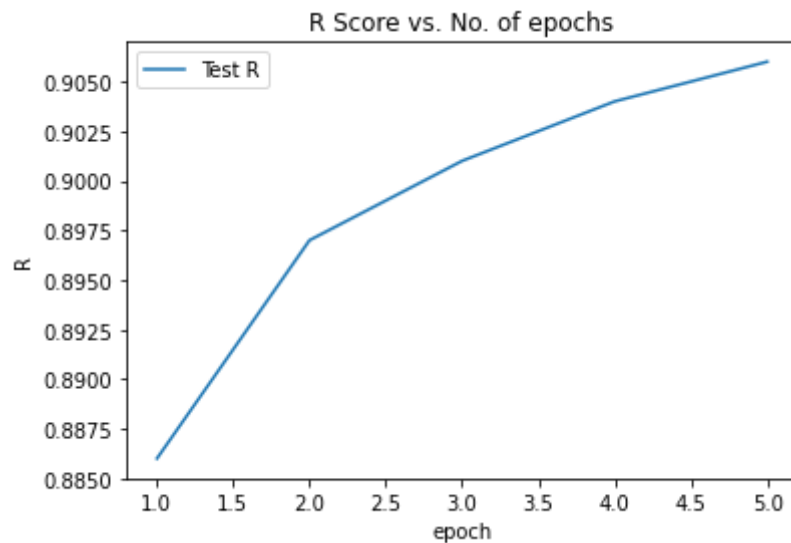
```
plt.plot(metrics['epoch'], metrics['Train PPL'], label="Train PPL")
plt.plot(metrics['epoch'], metrics['Valid PPL'], '-x', label="Test PPL")
plt.xlabel('epoch')
plt.ylabel('ppl')
plt.legend()
plt.title('Perplexity vs. No. of epochs');
```



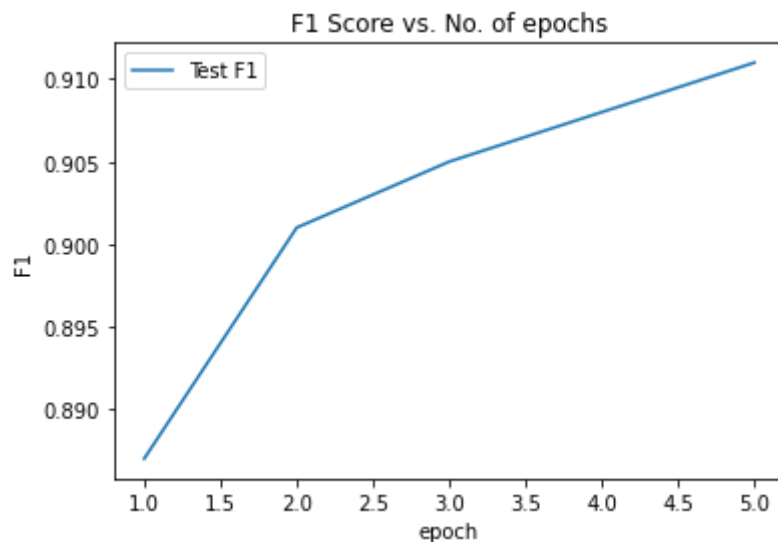
```
plt.plot(metrics['epoch'], metrics['Valid P'], label="Test P")
plt.xlabel('epoch')
plt.ylabel('P')
plt.legend()
plt.title('P Score vs. No. of epochs');
```



```
plt.plot(metrics['epoch'], metrics['Valid R'], label="Test R")  
plt.xlabel('epoch')  
plt.ylabel('R')  
plt.legend()  
plt.title('R Score vs. No. of epochs');
```



```
plt.plot(metrics['epoch'], metrics['Valid F1'], label="Test F1")  
plt.xlabel('epoch')  
plt.ylabel('F1')  
plt.legend()  
plt.title('F1 Score vs. No. of epochs');
```



Conclusion

We can see that in general, the scores increase, which means that our model is learning and improving

✓ 0s completed at 11:59 AM ● ✕