paper_code_documentation

Keli Chiu

03/10/2020

Paper: Text Infilling

Published date: January 18, 2019

Authors: Wanrong Zhu, Zhiting Hu, Eric P. Xing

Affiliation: Peking University, Carnegie Mellon University, Petuum Inc.

Code URL: https://github.com/VegB/Text_Infilling

Result

Not able to follow through the code tutorial.

Steps:

1. Download the code and install Texar.

```
git clone https://github.com/VegB/Text_Infilling
cd Text_Infilling
pip install --user -e .
```

2. Navigate to the infilling demo directory and download the dataset. The instruction provided on Github didn't work, I followed the `README.MD` in the `text_infilling` directory from this point. Downloaded data is stored in `yelp_data/`.

```
cd text_infilling
python data_utils.py
```

You might have to run extra code to install required packages if you don't have them already. In my case, I had to get `tensorflow` and `matplotlib`.

```
pip install "tensorflow>=1.15,<2.0"
pip install matplotlib
```

3. Code template to execute is provided as follow:

```
python [MODEL].py --mask_rate [MASK_RATE] --blank_num [BLANK_NUM]
  --filename_prefix 'pos.' --data_dir './yelp_data/pos/'
```

Here is what I tried, set mask rate to 30%, number of blank in an example is 1, and chose the self-attention mode.

```
python self_attn.py --mask_rate 30 --blank_num 1 --filename_prefix 'pos.'
  --data_dir './yelp_data/pos/'
```

An Error was raised from the above code. The error message:

```
Traceback (most recent call last):
  File "self_attn.py", line 335, in <module>
    tf.app.run(main=_main)
  File "/Users/kelichiu/Desktop/project_tif/lib/python3.7/site-packages/tensorflow_core/python/platform/
  line 40, in run
    _run(main=main, argv=argv, flags_parser=_parse_flags_tolerate_undef)
  File "/Users/kelichiu/Desktop/project_tif/lib/python3.7/site-packages/absl/app.py",
  line 300, in run
    _run_main(main, args)
  File "/Users/kelichiu/Desktop/project_tif/lib/python3.7/site-packages/absl/app.py",
  line 251, in _run_main
    sys.exit(main(argv))
  File "self_attn.py", line 64, in _main
    hparams=args.word_embedding_hparams)
  File "/Users/kelichiu/Desktop/repos/Text_Infilling/texar/modules/embedders/embedders.py",
  line 53, in __init__
    self._hparams)
  File "/Users/kelichiu/Desktop/repos/Text_Infilling/texar/modules/embedders/embedder_base.py",
  line 33, in _init_parameterized_embedding
    hparams, init_value, num_embeds, self.variable_scope)
  File "/Users/kelichiu/Desktop/repos/Text_Infilling/texar/modules/embedders/embedder_utils.py",
  line 191, in get_embedding
    initializer = layers.get_initializer(hparams["initializer"])
  File "/Users/kelichiu/Desktop/repos/Text_Infilling/texar/core/layers.py",
  line 333, in get_initializer
    initializer = utils.get_instance(hparams["type"], kwargs, modules)
  File "/Users/kelichiu/Desktop/repos/Text_Infilling/texar/utils/utils.py",
  line 209, in get_instance
    (class_.__module__, class_.__name__, key, class_args))
ValueError: Invalid argument for class
  tensorflow.python.ops.init_ops.RandomNormal: mean, valid args:set()
```

I have change the –mask_rate 30 and –blank_num 1 with variations between int, float and percentage, but no value could successfully run the code. Googled solutions but no relevant answers were found.

Paper: Enabling Language Models to Fill in the Blanks

Published date: September 10, 2020

Authors: Chris Donahue, Mina Lee, Percy Liang

Affiliation: Stanford University

Code URL: https://github.com/chrisdonahue/ilm

Result

Not able to follow through the code tutorial from Github. Google Collab code generates demo successfully.

Steps for Github repo

1. Download the code and initialize

```
git clone git@github.com:chrisdonahue/ilm.git
cd ilm
pip install -r requirements.txt
python -c "import nltk; nltk.download('punkt')"
pip install -e .
```

You might have to run extra code to install required packages if you don't have them already. In my case, I had to get `nltk`.

```
pip install nltk
```

2. The following script was provided to download training dataset and create training example. It requires us saving the script as an sh file in order to execute it. I saved it in the repo root directory as script.sh.

```
DATASET=arxiv_cs_abstracts

pushd data
./get_${DATASET}.sh
popd

for SPLIT in train valid
do
    python create_ilm_examples.py \
        ${SPLIT} \
        data/char_masks/${DATASET} \
        --seed 0 \
        --data_name ${DATASET} \
        --data_split ${SPLIT}
done
```

3. Execute the script by `./script.sh`. If permission error is displayed, try `chmod +x script.sh` then execute the script again:

```
# chmod +x script.sh
./script.sh
```

3.1. If the above execution raises an error that tells you that `arxiv_cs_abstracts.txt` is not found, go to this link to download the missing file: https://docs.google.com/uc?export=download&id=1N3MbvpgZAmNgiZgnpXAQFzHrU7Tt3Blb

3.2. Unzip the file and rename it to `arxiv_cs_abstracts.txt`, move it to `repo/data/raw_data/arxiv_cs_abstracts/`

4. The following script was provided to train the model (fine-tune GPT-2) by the training examples. It requires us saving the script as an sh file in order to execute it. I saved it in the repo root directory as training.sh. Execute it by `./training.sh`:

```
DATASET=arxiv_cs_abstracts
TRAIN_DIR=train
EXAMPLES_DIR=data/char_masks/${DATASET}
python train_ilm.py \
    experiment_${DATASET} \
    ${TRAIN_DIR} \
    ${EXAMPLES_DIR} \
    --seed 0 \
    --train_examples_tag train \
    --eval_examples_tag valid \
    --eval_max_num_examples 512
```

The fine-tuning will start. According to the paper, it will take 1 day to complete with GPU. For a computer that doesn't have a GPU, it will take longer.

Google Colab

Link: https://colab.research.google.com/drive/1So95M0hHefyNm_eELglCna_ZayoDX6KV?usp=sharing

Input

```
English Class

Christie was good at doing _. _ _ _ She ended up failing the test.
```

Encode

```
English Class

Christie was good at doing<|infill_word|>.<|infill_paragraph|><|infill_sentence|>
  <|infill_sentence|> She ended up failing the test.
```

Output

```
English Class

Christie was good at doing things. One day at the library she had to take a math test.
She made a terrible grade. Christie felt terrible and did it anyways.
She ended up failing the test.

English Class

Christie was good at doing homework. She would stay up late to study.
When she was home for class, she would go into and out of class.
But her teacher couldn't find her anywhere. She ended up failing the test.
```