NLP Final Project writeup plan

Status since check in:

In my original project plan, I wanted to apply dataset cartography onto the SQuAD. The model was successfully trained on the SQuAD data set. However, during evaluation, the model constantly crashed at the end of evaluation. It was then that I discovered the provided SQuAD evaluation dataset did not have offset mappings to evaluate the dataset.

Hence, the project’s direction was shifted to now focus on the SNLI Natural Language Inference dataset instead. The project will still use dataset cartography to separate data, and examine effectiveness of curriculum learning.

Base untrained electra model:

Evaluation results:

{'eval\_loss': 1.09840726852417, 'eval\_accuracy': 0.33753302693367004, 'eval\_runtime': 237.1615, 'eval\_samples\_per\_second': 41.499, 'eval\_steps\_per\_second': 5.191}

Baseline: Checkpoint-11000 model:

Evaluation results:

{'eval\_loss': 0.47822514176368713, 'eval\_accuracy': 0.8303190469741821, 'eval\_runtime': 233.4085, 'eval\_samples\_per\_second': 42.166, 'eval\_steps\_per\_second': 5.274}

Baseline:

Cartography dataset is downloaded from Swayamdipta et al., 2020. Although we will be using their values, it is to be noted that the models used are different. In our instance, we are using Electra-small model, and in the paper, they use RoBERTa-large. However, it is generalizable as both models are based on BERT transformer models, and RoBERTa-large is has more parameters, which suggests that it is at least as powerful and generalizable as electra-small. This means that the data cartography labels are at least its own value, or worse.

To run curriculum learning on 22000 examples:

python3 run.py --do\_train --task nli --dataset snli --output\_dir ./trained\_model\_nli/ --max\_train\_samples 22000 --curr y --carto\_dataset ../cartography/data/data\_map\_coordinates/snli\_roberta\_0\_6\_data\_map\_coordinates.jsonl

Based on the curriculum learning paper, they suggested started with a single training pass on the easy difficulty examples, before a second pass now including an equal amount of medium difficulty examples, and finally a last pass with the hard examples.

Cartography mapping:

easy to learn (correctness= 5 to 6)-> easy,

ambiguous ( correctness = 3 to 4-> medium,

hard to learn (correctness = 0 to 2) -> hard

The curriculum plan is generated by dividing the total examples to be trained by 6 to get the curriculum increment. Then we incrementally add this increment’s number of increasing difficulty into our dataset, and retrain the model with this new dataset

In this case, we use 22000 examples:

Curriculum plan:

3666 easy examples for 3 epochs

7332 of easy and medium examples for 3 epochs

11000 of easy, medium, and hard examples for 3 epochs

Total 8253 steps

Normal Baseline of 22000 examples

All difficulties of 22000 in 3 epochs

Total 8250 steps