Aluma Etzion
211444179

Advanced NLP Exercise 1

Open Questions

1.

- Quoref- this dataset requires resolving complicated coreference and identifying correctly the different entities. This measures coreferential reasoning.
- EntQA- in this dataset, the model needs to select the right entity from a database given a question or a mention. By that, it mesaures entity linking.
- TACRED- this dataset contains examples of above 40 relation types, and it requires solving relation classification.

2. (a)

- Self-consistency

In this method, multiple solutions are generated independently, and the most consistent answer given by the highest number of solution chains is returned as the final answer.

Advantages- increases test-time compute, and can be used even without verifiers.

Computational bottlenecks- the whole generation process has to be done several times, heavily using the generative model and utilizing GPUs. It also requires keeping the answer in memory.

Parallelization- possible, because each solution chain is independent from the others. The model can be run in parallel for each chain.

- Using verifiers to select the final answer-

This method is similar to self-consistency, but instead of choosing the most consistent answer, all answers are first verified, and the answer will be the most consistent among the verified answers or just the best answer based on verification.

Advantages- adds an additional layer of confidence to the final answer given.

Computational bottlenecks- similar to self-consistency, the whole generation process heavily utilizes GPUs, it also requires keeping the answer in memory. Verifying is usually not heavy, using things like unit-testing.

Parallelization—As earlier, the model can be run separately for each chain in parallel. Also, verifying the answers can be done in parallel for each answer separately.

- Generating parts of chains, and selecting the best ones using verifiers-

Here, parts of chains are generated and verified, so that only the best ones are chosen to continue generating from.

Advantages- verifying the chains early on saves computational effort from being wasted on bad solutions from the start.

Computational bottlenecks- running the model to generate all of the chains and keeping chains, which is heavier than saving only the final answer.

Parallelization- partially possible. Several parts of the chains can be generated in parallel, but to get to the final answer, we must stop and verify.

(b) If I only have one GPU, I can’t run the model in parallel. Thus, I choose to generate part of the chains, check them, and select the best ones. I have a memory to store the good chain found so far and more information that would help me utilize the GPU I have for promising chains.

Programming Exercise- github link

- In my search, the best configuration (lr=0.0001, epoch num=2, batch size=16) was best for both validation and test sets.

  When I looked for a pattern in sentences that the best configuration got correctly but the worst didn’t, I found that when two sentences began with the same n-gram but then changed, the worst configuration model didn’t catch the fact that they are equivalent. Also in other cases where the same words were used but in a different order the worst model failed. Some examples:

  Sentence 1: The daily Hurriyet said the raid aimed to foil a Turkish plot to kill an unnamed senior Iraqi official in Kirkuk .
  Sentence 2: The daily Hurriyet said the raid aimed to foil a Turkish plot to kill an unnamed senior Iraqi Kurdish official in Kirkuk , but Gul has denied any Turkish plot .
  True Label: 0, Best configuration prediction: 0, Worst configuration prediction: 1

  Sentence 1: " The vulnerabilities all relate to a lack of effective FAA oversight that needs to be improved , " the report said .
  Sentence 2: " These vulnerabilities all relate to a lack of effective FAA oversight and , if not corrected , could lead to an erosion of safety , " said the report .
  True Label: 0, Best configuration prediction: 0, Worst configuration prediction: 1

  Best configuration- lr=0.0001, epoch num=2, batch size=16, worse confifuration- lr=1e-05, epoch num = 3, batch size=32.