End-to-End finetuning of RAG (including DPR retriever) for Question Answering.

This finetuning script is actively maintained by Shamane Siri. Feel free to ask questions on the Forum or post an issue on GitHub and tag @shamanez.

Others that helped out: Patrick von Platen (@patrickvonplaten), Quentin Lhoest (@lhoestq), and Rivindu Weerasekera (@rivinduw)

The original RAG implementation is able to train the question encoder and generator end-to-end. This extension enables complete end-to-end training of RAG including the context encoder in the retriever component. Please read the accompanying blog post for details on this implementation.

The original RAG code has also been modified to work with the latest versions of pytorch lightning (version 1.2.10) and RAY (version 1.3.0). All other implementation details remain the same as the original RAG code. Read more about RAG at https://arxiv.org/abs/2005.11401.

This code can be modified to experiment with other research on retrival augmented models which include training of the retriever (e.g. REALM and MARGE).

To start training, use the bash script (finetune_rag_ray_end2end.sh) in this folder. This script also includes descriptions on each command-line argument used.

Note

⚠ This project should be run with pytorch-lightning==1.3.1 which has a potential security vulnerability

Testing

The following two bash scripts can be used to quickly test the implementation.

1. sh ./test_run/test_rag_new_features.sh
    - Tests the newly added functions (set_context_encoder and set_context_encoder_tokenizer) related to modeling rag.
    - This is sufficient to check the model's ability to use the set functions correctly.
2. sh ./test_run/test_finetune.sh script
    - Tests the full end-to-end fine-tuning ability with a dummy knowlendge-base and dummy training dataset (check test_dir directory).
    - Users can replace the dummy dataset and knowledge-base with their own to do their own finetuning.

Comparison of end2end RAG (including DPR finetuning) VS original-RAG

We conducted a simple experiment to investigate the effectiveness of this end2end training extension using the SQuAD dataset. Please execute the following steps to reproduce the results.

- Create a knowledge-base using all the context passages in the SQuAD dataset with their respective titles.

- Use the question-answer pairs as training data.
- Train the system for 10 epochs.
- Test the Exact Match (EM) score with the SQuAD dataset's validation set.
- Training dataset, the knowledge-base, and hyperparameters used in experiments can be accessed from here.

Results

- We train both models for 10 epochs.

Model Type	EM-Score
RAG-original	28.12
RAG-end2end with DPR	40.02