Scripts for evaluation

There are six python scripts to validate, evaluate, and characterize Task 1 and Task 3 runs using annotations and assessments for ARQMath–3 Task 3:

1. `produce_task3_baseline.py` produces the GPT–3 baseline run for ARQMath–3 Task 3.
2. `validate_task3_results.py` validates Task 3 runs.
3. `evaluate_task1_results_manual.py` evaluates Task 1 runs using manual evaluation measures for ARQMath–3 Task 3.
4. `evaluate_task3_results_manual.py` evaluates Task 3 runs using manual evaluation measures for ARQMath–3 Task 3.
5. `evaluate_task3_results_automatic.py` evaluates Task 3 runs using automatic evaluation measures for ARQMath–3 Task 3.
6. `evaluate_task3_results_postevaluation.py` characterizes Task 3 runs using post–evaluation assessment for ARQMath–3 Task 3.

All scripts have been type–checked and tested with Python 3.7, 3.8, and 3.9.

`produce_task3_baseline.py`

This script can be used to produce the GPT–3 baseline run for ARQMath Task 3. A number of python libraries are required by the script:

```
pip install openai==0.18.1
pip install git+https://github.com/MIR-MU/pv211-utils.git@1.1.6
```

Furthermore, you need to set the `OPENAI_API_KEY` environmental variable to contain your API key for OpenAI cloud services, see https://beta.openai.com/account/api–keys:

```
export OPENAI_API_KEY=123456789
```

Two inputs should be provided as in the example command shown below:

```
python3 produce_task3_baseline.py
  -out "Baseline2022-task3-GPT3-auto-both-generate-P.tsv"
  -year 2022
```

1. The path to the Task 3 run.
2. The year from which we take our topics for Task 1 and 3. Valid values are `2020`, `2021`, and `2022`.

`validate_task3_results.py`

This script can be used to validate runs for the submissions in Task 3. One input should be provided as in the example command shown below:

```
python3 validate_task3_results.py
  -in "Baseline2022-task3-GPT3-auto-both-generate-P.tsv"
```

1. The path to the Task 3 run.

`evaluate_task1_results_manual.py`

This script can be used to evaluate Task 1 runs using manual evaluation measures for ARQMath–3 Task 3 (AR and P@1). Three inputs should be provided as in the example command shown below:

```
python3 evaluate_task1_results_manual.py
  -in "baseline_tangents_task1_2022.tsv"
  -excluded_topics '[]'
  -qrel "qrel_task1_2022_official.tsv"
```

1. The path to the Task 1 run.
2. A JSON array of Task 1 and 3 topics excluded from the evaluation.
3. The path to a file with relevance judgements for ARQMath Task 1. You can find this file in directory `../Task 1/Qrel Files`.

evaluate_task3_results_manual.py

This script can be used to evaluate Task 3 runs using manual evaluation measures for ARQMath−3 Task 3 (AR and P@1). Four inputs should be provided as in the example command shown below:

```
python3 evaluate_task3_results_manual.py
  -in "Baseline2022-task3-GPT3-auto-both-generate-P.tsv"
  -excluded_topics '[]'
  -map "teams_answer_id.tsv"
  -qrel "qrel_task3_2022_official_complete.tsv"
```

1. The path to the Task 3 run.
2. The path to a file that maps topic IDs and run names to synthetic answer IDs for ARQMath Task 3. You can find this file in directory `Data Files`.
3. A JSON array of Task 1 and 3 topics excluded from the evaluation.
4. The path to a file with complete relevance judgements (including 5: system failure and 6: do not know judgements) for ARQMath Task 3. You can find this file in directory `Qrel Files`.

evaluate_task3_results_automatic.py

This script can be used to evaluate Task 3 runs using automatic evaluation measures for ARQMath−3 Task 3 (LO and CS). A number of python libraries are required by the script:

```
pip install lxml beautifulsoup4 transformers>=4.20.0 bert-score==0.3.11
pip install git+https://github.com/MIR-MU/ARQMathCode.git
```

Eleven inputs should be provided as in the example command shown below:

```
python3 evaluate_task3_results_automatic.py
  -all_task1_answers "task1_arqmath3_runs/"
  -all_task3_answers "task3_arqmath3_runs/"
  -excluded_task1_run_ids '[]'
  -excluded_task3_run_ids '["GPT3"]'
  -excluded_topics '[]'
  -use_task1_answers true
  -collection "collection/"
  -in "Baseline2022-task3-GPT3-auto-both-generate-P.tsv"
  -map "teams_answer_id.tsv"
  -task1_qrel "qrel_task1_2022_official.tsv"
  -task3_qrel "qrel_task3_2022_official_complete.tsv"
```

1. The path to a directory with all runs for ARQMath−3 Task 1. You can find these in directory `../Task 1/Data Files/All Task 1 runs`.
2. The path to a directory with all runs for ARQMath−3 Task 3. You can find these in directory `Data Files/All Task 3 runs`.

3. A JSON array of Task 1 run ids from the same team as the run that is being evaluated. Should be empty for new systems from teams that did not participate in ARQMath–3 Task 3.
4. A JSON array of Task 3 run ids from the same team as the run that is being evaluated. Should be empty for new systems from teams that did not participate in ARQMath–3 Task 3.
5. A JSON array of Task 1 and 3 topics excluded from the evaluation.
6. Whether the script should use relevant answers for ARQMath–3 Task 1 in the evaluation rather than just relevant answers for ARQMath–3 Task 3. Valid values are `true` and `false`.
7. The path to a directory with the ARQMath collection V1.3. You can find it in directory `../../../../Collection`.
8. The path to the Task 3 run.
9. The path to a file that maps topic IDs and run names to synthetic answer IDs for ARQMath Task 3. You can find this file in directory `Data Files`.
10. The path to a file with relevance judgements for ARQMath Task 3. You can find this file in directory `../Task 1/Qrel Files`.
11. The path to a file with complete relevance judgements (including 5: system failure and 6: do not know judgements) for ARQMath Task 3. You can find this file in directory `Qrel Files`.

evaluate_task3_results_postevaluation.py

This script can be used to characterize Task 3 runs using post–evaluation assessment for ARQMath–3 Task 3 (MG and UI). Four inputs should be provided as in the example command shown below:

```
python3 evaluate_task3_results_postevaluation.py
  -in "Baseline2022-task3-GPT3-auto-both-generate-P.tsv"
  -excluded_topics '["A.367"]'
  -map "teams_answer_id.tsv"
  -qrel "task3-extra-assessment.tsv"
```

1. The path to the Task 3 run.
2. A JSON array of Task 1 and 3 topics excluded from the evaluation.
3. The path to a file that maps topic IDs and run names to synthetic answer IDs for ARQMath Task 3. You can find this file in directory `Data Files`.
4. The path to a file with post–evaluation assessment for ARQMath–3 Task 3. You can find this file in directory `Qrel Files`.