**평가지표 기반 간단 알고리즘 개발**

**산출물 정리 & 실행 방법 설명**

**1. 개요**

평가지표 5개에 대해 모두 python으로 구현하였습니다. 개발 후 테스트에 사용한 python 버전은 3.12.3입니다. 전부 python 내장 라이브러리만 사용하였으므로 python 버전이 너무 낮지 않으면 문제가 되지 않을 것으로 보이긴 하지만, 가급적 python 3.12.3 으로 실행해 주시는 것이 좋을 것 같습니다.

입력/출력 문서 형식의 경우 개발 시 참조했던, notion 문서의 파일 형식을 그대로 맞추었습니다. 이 문서에서는 별도로 설명하지는 않습니다.

**2. 소스 각 폴더별 설명**

|  |  |  |
| --- | --- | --- |
| 폴더 | 파일명 | 설명 |
| ./01 | marker\_process.py | 지표 1 구현 |
| run\_10.sh | 지표 1 10회 실행 쉘 스크립트 |
| ./02 | user\_feature\_build.py | 지표 2 구현 |
| run\_20.sh | 지표 2 10회 실행 쉘 스크립트 |
| ./03 | matcher\_eval.py | 지표 3 구현 |
| ./04 | user\_theme\_eval.py | 지표 4 구현 소스 |
| ./05 | message\_eval.py | 지표 5 구현 소스 |

**3. 각 지표 별 실행 방법**

**(1) 지표 1**

./01 폴더로 이동하여 다음을 실행합니다.

$ python3 marker\_process.py --input sample.json --output result.json --log run1.log

이 때, ./01 폴더에 sample.json 파일이 있어야 하며, 요구조건과 동일한 형식의 result.json 과 run1.log 파일이 생성됩니다.

10번실행을 위해 run\_01.sh 파일을 만들어 놓았습니다.

1) ./01 폴더로 이동하세요.

2) 다음을 실행하세요.

$ chmod +x run\_01.sh

3) 그 다음 run\_01.sh를 실행하시면 10번 수행됩니다.

$ run\_01.sh

이 run\_01.sh 에는 다음과 같이 10번 반복실행하는 코드가 작성되어 있습니다.

for i in {1..10}; do

python3 marker\_process --input sample.json --output result\_$i.json --log run\_$i.log

done

**(2) 지표 2**

./02 폴더로 이동하여 다음을 실행합니다.

$ python3 user\_feature\_build.py --input users\_logs.json --days 30 --output result.json --log run.csv

이 때 ./02 폴더에 users\_logs.json 파일이 있어야 합니다.

10번 실행을 위해 ./02 폴더에 run\_02.sh 파일을 만들어 놓았습니다.

1) ./02 폴더로 이동하세요.

2) 다음을 먼저 실행하세요.

$ chmod +x run\_02.sh

3) 그 다음 run\_02.sh 를 실행하시면 10번 수행됩니다.

$ run\_02.sh

이 run\_02.sh 는 다음과 같이 실행되도록 구현되어 있습니다.

for i in {1..10}; do

python3 user\_feature\_build.py --input sample\_user.json --days 30 \

--output feature\_u\_demo1\_$i.json --log run\_$i.csv

Done

**(3) 지표 3**

./03 폴더로 이동하여 다음을 실행합니다.

$ python3 matcher\_eval.py \

--catalog catalog.json \

--scenarios scenarios.json \

--answers answers.json \

--topk 5 \

--weights "theme=3.0,tag=0.8,dist=0.10" \

--output recsult. json \

--log summary.csv

이 때, ./03 폴더에 catalog.json, scenarios.json, answers.json 파일이 있어야 합니다.

**(4) 지표 4**

./04 폴더로 이동하여 다음을 실행합니다.

$ python3 user\_theme\_eval.py \

--themes themes.json \

--users user\_profiles.json \

--answers answers.json \

--topk 5 \

--weights "cat=2.0,tag=1.0,region=1.5" \

--output result. json \

--log summary.csv

이 때, ./04 폴더에 themes.json, user\_profiles.json, answers.json 파일이 있어야 합니다.

**(5) 지표 5**

./05 폴더로 이동하여 다음을 실행합니다.

$ python3 message\_eval.py \

--catalog message\_catalog.json \

--scenarios scenarios.json \

--answers answers.json \

--weights "theme=2.0,tag=1.0,region=1.5,recency=0.5" \ --topk 1 \

--output result.json \

--log summary.csv

이 때 ./05 폴더에 message\_catalog.json, scenarios.json, answers.json 파일이 있어야 합니다.