CS 410 FA24: How to use the software

Ad recommendation with sentiment analysis

Junyoung Lee(jl298, Coordinator)

All code in this package is written in Python and Bash, and has been tested on WSL2 Ubuntu 22.04.5 LTS. Due to the github's size limitation, I uploaded a randomly generated dataset for the demo. You can download the actual dataset from the website following if you want to test with one of them. Dataset link

Setup

1. Download and unzip the cs410.project.zip file from the link below:
    - https://github.com/jl298/cs410-project/archive/refs/heads/main.zip

2. Navigate to the src directory and install required Python packages:
    - `$ cd src`
    - `$ pip install -r ./requirements.txt`

3. All subsequent scripts should be run from this `/src` directory.

Training Steps

1. Run the following script to perform the ingestion step that converts Amazon review data into SNS post data:
    - `$ python ./ingestion_backfill.py`

2. Verify that the following 4 files have been created:
    - `$ ls ./dataset/`
        1. `sns-user-dataset.json`
        2. `sns-posts-dataset*.json`
        3. `sns-posts-dataset.recommender.json`
        4. `sns-posts-dataset.sentiment.json`

3. Run the following script to train the sentiment analysis model using data partitions:
    - `$ python ./sentiment_training.py`

4. Verify that the model directory and model files have been created:
    - `$ ls ./model/sns_sentiment_model/`

5. Run the following script to apply the sentiment analysis model to the recommender data partition to create a sentiment dataset:
    - `$ python ./sentiment_backfill.py`

6. Verify that the following file has been created:
    - `$ ls ./dataset/sentiment-dataset.json`

7. Train the recommender model using the sentiment dataset created for the recommender:
    - `$ python ./recommender_training.py`

8. Verify that the model directory and model files have been created:
    - `$ ls ./model/recommender_model/`

Posting a Comment via SNS UI

1. You can run a command-line version of the SNS UI using the script below. After execution, you'll see a prompt for entering commands. The user for the demo is randomly selected from the dataset, and you can check user information with the user command. You can also directly open the `../dataset/sns-user-dataset.json` file to view all information.
    - `$ python ./run.py`
    - `> user`

2. You can see a list of available commands by typing help at the SNS prompt:
    - `> help`

3. The feed command allows you to browse all available advertisement posts through pagination. Since this is a demo SNS, there's no feature to hide previously read posts, and you can always view all posts again. Each SNS post is an advertisement for one product.
    - `> feed`
    - `> feed {PAGE_NUMBER}`
    - `> feed next`
    - `> feed prev`

4. After finding an advertisement post you like through the feed command, use the comment command to leave your thoughts about the product, and for demo convenience, manually enter your reaction.
    - ```
      > comment
      Enter post ID: 0
      Enter comment text: I love this product.
      Enter reaction (liked/viewed/ignored): liked
      ```

5. After leaving a comment, exit the SNS UI using the quit command.
    - `> quit`

Inference Steps

1. Run the following script to perform sentiment analysis on newly added comments for the corresponding product:
    - `$ python ./sentiment_inference.py`

2. Verify that new entries have been added:
    - If `jq` command is installed, use the following command to verify that sentiment data has been properly added:
      ```

        - `jq '.[] | select(.tag == "online" and .user_id == "'$(jq -r '.user_id' ../dataset/sns-user-dataset.json)'")' ../dataset/sentiment-dataset.json`
    - If `jq` command is not installed, open `../dataset/sns-user-dataset.json` to check the user_id, then directly verify in ../dataset/sentiment-dataset.json if there are JSON entries with that user_id and tag item as "online"
3. Get product recommendations from the recommendation system using the command:
    - `$ python ./recommender_inference.py`

4. Verify that 5 recommended products have been output:
    - `$ cat ../dataset/recommendations-dataset.json`

5. Run the SNS UI and use the update command to create a new advertisement post with the top recommendation:
    - `$ python ./sns.py`
    - `> update`

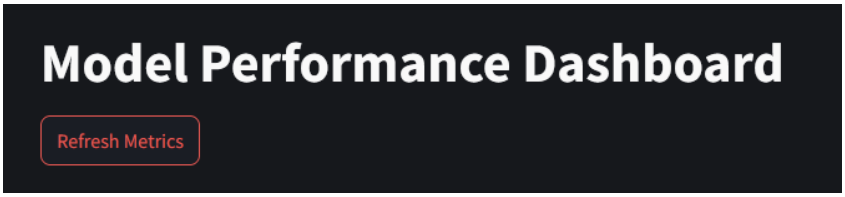
6. Check the last page using the SNS feed command, then move to specific pages using the feed command with page numbers to find the newly created advertisement post and its post ID:
    - `> feed`
    - `> feed {page_number}`

7. Use the SNS click command to click on that advertisement:
    - ```
      > click
      Enter post ID: {post_id}
      ```

Confirming Stats

1. Running the following bash command will launch a web app created with the streamlit package in your terminal. You can access the stats UI by clicking the output URL or directly entering http://localhost:8501 in your web browser:
    - `$ ./run_stats.sh`

2. If nothing appears, click the "Refresh Metrics" button as shown in the screenshot below.


      ```