1. Problem

This task benchmarks recommendation with implicit feedback on the MovieLens 20 Million (ml-20m) dataset with a Neural Collaborative Filtering model.
The model trains on binary information about whether or not a user interacted with a specific item.

2. Directions

Steps to configure machine

From Source

1. Install MXNet(CPU or GPU)
2. Install `unzip` and `curl`

```
sudo apt-get install unzip curl
```

3. Checkout the johnsonkee repo

```
git clone http://github.com/johnsonkee/recommend.git
```

4. Install other python packages

```
cd recommendation
pip install -r requirements.txt
```

From Docker

1. Checkout the johnsonkee repo

```
git clone http://github.com/johnsonkee/recommend.git
```

2. Install CUDA and Docker

```
source reference/install_cuda_docker.sh
```

3. Get the docker image for the recommendation task

```
# Pull from Docker Hub
docker pull mxnet/python:1.2.0_gpu_cuda9
```

Steps to download and verify data

From Source

You can download and verify the dataset by running the `download_dataset.sh` and `verify_dataset.sh` scripts in the parent directory. Before running the following codes, make sure you are in *`recommend`* directory:

```
# Creates ml-20.zip
./download_dataset.sh
# Confirms the MD5 checksum of ml-20.zip
./verify_dataset.sh
```

From Docker

After pulling the image `mxnet/python:1.2.0_gpu_cuda9`, you can continue the following codes.

1. Build a container through the image

```
nvidia-docker run --name johnsonkee_mxnet -ti \
mxnet/python:1.2.0_gpu_cuda9 /bin/bash
```

2. Build a directory to start your workers

```
cd /home
```

3. Checkout the johnsonkee repo

```
git clone http://github.com/johnsonkee/recommend.git
```

4. Download and verify dateset

```
# Creates ml-20.zip
cd recommend
./download_dataset.sh
# Confirms the MD5 checksum of ml-20.zip
./verify_dataset.sh
```

Steps to run and time

From Source

Run the `run_and_time.sh` script with an integer seed value between 1 and 5

```
./run_and_time.sh SEED
```

From Docker

Run the `run_and_time.sh` script with an integer seed value between 1 and 5

```
# make sure you are in the `recommend` directory
./run_and_time.sh SEED
```

3. Dataset/Environment

Publication/Attribution

Harper, F. M. & Konstan, J. A. (2015), 'The MovieLens Datasets: History and Context', ACM Trans. Interact. Intell. Syst. 5(4), 19:1--19:19.

Data preprocessing

1. Unzip
2. Remove users with less than 20 reviews
3. Create training and test data separation described below

Training and test data separation

Positive training examples are all but the last item each user rated.
Negative training examples are randomly selected from the unrated items for each user.

The last item each user rated is used as a positive example in the test set.
A fixed set of 999 unrated items are also selected to calculate hit rate at 10 for predicting the test item.

Training data order

Data is traversed randomly with 4 negative examples selected on average for every positive example.

4. Model

Publication/Attribution

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu and Tat-Seng Chua (2017). Neural Collaborative Filtering. In Proceedings of WWW '17, Perth, Australia, April 03-07, 2017.

The author's original code is available at hexiangnan/neural_collaborative_filtering.

5. Quality

Quality metric

Hit rate at 10 (HR@10) with 999 negative items.

Quality target

HR@10: 0.6289

Evaluation frequency

After every epoch through the training data.

Evaluation thoroughness

Every users last item rated, i.e. all held out positive examples.

6. About

This project was rewritten from mlperf'recommendation by Xianzhuo Wang when Xianzhuo Wang was an intern a Cambricon.

The major difference between the two is that the original one uses `PyTorch` as framework while the new one uses `MXNet` as framework. In addition, the new one can support for two new datasets:
ml-latest-small
ml-latest

7. Issues & Suggestions

If you have any questiones, contact me 876688461@qq.com or creat an issue