**Advanced Topics in Recommendation Systems – EX2**

Authors:

Maya Vilenko - 315339747

Lior Tondovski – 307870766

Ilan Vasilevsky - 322545682

**Parts 1-3 – Model Objective:**

The model’s objective is to predict which future item a user will likely consume next. The objective function measures the log-loss (Binary Cross Entropy). The model will be trained on positive samples for each user and negative samples (the negative samples drawing mechanism Is explained in the chapter 6).

**Objective function’s components notations and definitions:**

The objective function:

**The update step for each parameter:**



**4. Pseudo Code:**

*Preprocessing: Drawing Negative Samples for each user (full explanation on the way we did it is the next section.*

For Epoch in Epochs:

For each sample in train set (User, Item, Binary Score):

Prediction =

Update the model parameters:



objective\_score=0

For each sample in the validation set (User, Item, Binary Score):

Prediction =

objective\_score

Validation\_objective\_score = objective\_score/len(validation\_samples)

If stopping criterion is met (early stopping):

Break the loop

\*\* The stopping criteria we chose is to stop the training loop if the validation objective score is increasing for 2 epochs in a row.

**5. Hyperparameters:**

* The size of the item and user embedding
* learning rate
* Stopping Criteria
* Number of Epochs
* The way of drawing the negative samples.
* Standard deviation for initialization of user embeddings
* Standard deviation for initialization of item embeddings

**6. Validation Set**

We must create the validation set to check for convergence using an evaluation metric/objective function. To create the validation set, we took 3 positive examples and drew 97 negative examples for each user, so that each user has 100 samples. We make sure that no item or user appears in the validation and doesn't appear in the training set.

Drawing Negative Examples:

The negative examples were drawn in two ways:

1. A random selection is made from the set of items that the user did not select/view.

2. A Drawing mechanism that considers how many times the item has been seen (item popularity).

We assumed at this point that the lottery considering item popularity would yield good results on both the test set in which the item the user did not select/view was randomly drawn and on the test set in which it was drawn based on popularity (we will check this later by different evaluation metrics).

The difference between Train & Test set:

* In the Train set, we drew as many negative samples as positive samples.
* In the validation set, we took 3 positive samples and 97 negative samples for each user, so that for each user, there are a total of 100 samples. This allows us to calculate metrics such as MPR and Hit Rate@k.

**7. Training the Best Model:**

In order to gain "the best model", a stopping criterion can be implemented in many ways. During training, we decided to stop if the objective function (on the validation set) increased two epochs in a row.

**The MPR was calculated as follows**:

We calculated the MPR by comparing each positive example for each user with the 97 negative examples. In the end, the MPR is an average of all.

**Hit-Rate@k calculation:**

By sorting all the samples for each user, we calculated how many positive samples are in a window containing k samples. To calculate the final score, all the users' hit rates are averaged, we used k=1, k=10, k=50 as requested.

**Results:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Train Sample Type | Validation  Sample Type | MPR | HR@1 | HR@10 | HR@50 | validation\_loss |
| random | random | 0.111 | 0.117 | 0.633 | 0.972 | 0.818 |
| random | popularity | 0.110 | 0.116 | 0.634 | 0.973 | 0.818 |
| popularity | random | 0.110 | 0.120 | 0.635 | 0.972 | 0.818 |
| popularity | popularity | 0.110 | 0.122 | 0.636 | 0.973 | 0.818 |

We Noticed that using embeddings that were trained on a train set consisting of negative samples based on popularity yields slightly better results on both validation sets – with negative samples based on popularity and random choice. Therefore we used those embeddings for predictions on the test set