Music Recommendation System

Ceng History X - Iteration Report 2

April 12, 2014

Team Members:

1) Asena Ok, 1746296, asenaok@gmail.com

2) Ayşe Aybüke Taşdirek, 1746353 aybuketasdirek@gmail.com

3) Birant Altınel, 1745744 birantaltinel@gmail.com

4) Hacer Nihal Tarkan, 1744291, tarkan.nihal@gmail.com

Tasks for Iteration 2:

Planning Tasks	**Responsible Person**	**Implementation**	**Is completed?**
Developing Content Based Recommendation	All Team	No	This method is considered as inapplicable for our project
Improvement of recommendation algorithm with Neighborhood-Based Collaborative	All Team	Yes	Yes
Defining Evaluation Module Metrics	All Team	Yes	Yes

Table 1: Tasks for second iteration

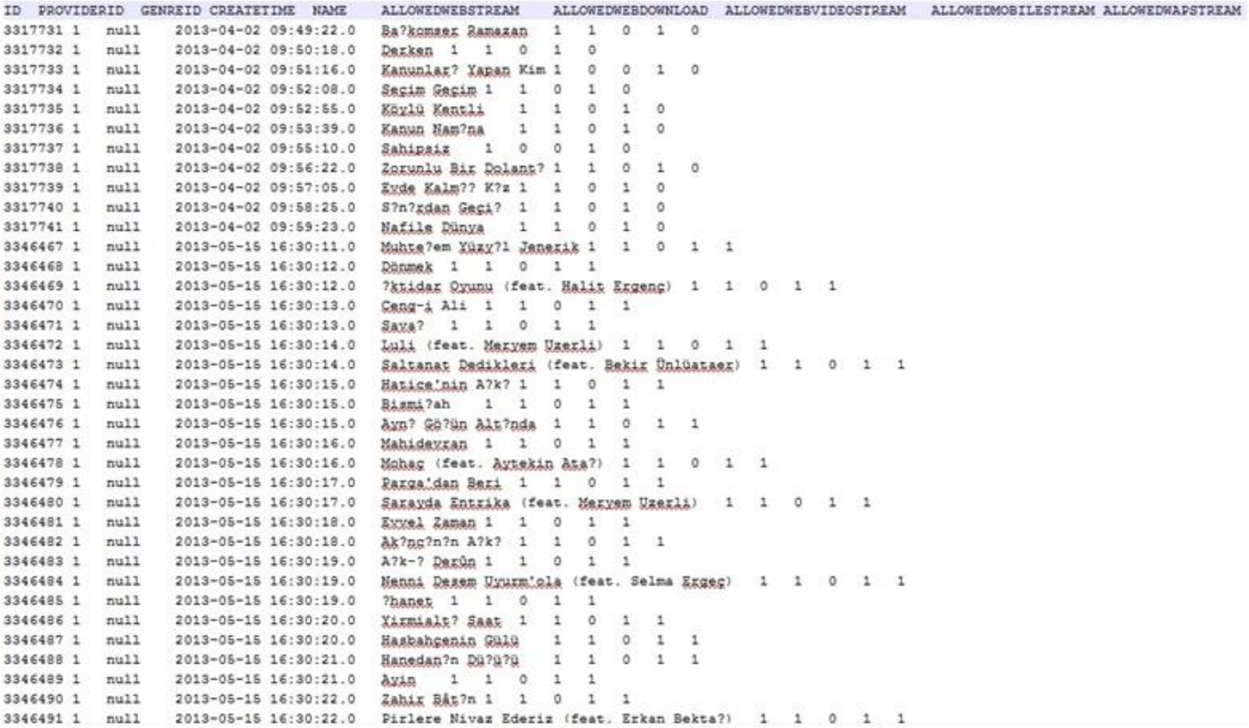
1. Why not Content-based?

In the first iteration, we've decided that tag based recommendation is not applicable for our data. Therefore, we analyzed the 'genre' of the songs to use while improving content-based approach. However, we just realized that 'genre' of the songs is not true. Please see the example below to observe the 'genres' are as irrelevant as we mentioned:

ID	PROVIDERID	GENREID	CREATETIME	NAME
3123440	7	22	*2012-03-15 16:51:58.0*	*Rolling In The Deep*
3383460	2	22	*2013-09-02 21:06:15.0*	*Ya Ali*
3383461	2	22	*2013-09-02 21:06:15.0*	*Erenler Cemi*
3383462	2	22	*2013-09-02 21:06:16.0*	*Kerbela'ya Yürüyorum*

Table 2: Song in the same genre

Also, nearly %10 of the songs has no genre information:



```
ID   PROVIDERID  GENREID CREATETIME   NAME       ALLOWEDWEBSTREAM     ALLOWEDWEBDOWNLOAD  ALLOWEDWEBVIDEOSTREAM    ALLOWEDMOBILESTREAM ALLOWEDWAPSTREAM
3317731 1   null    2013-04-02 09:49:22.0   Ba?komser Ramazan   1   1   0   1   0
3317732 1   null    2013-04-02 09:50:18.0   Derken  1   1   0   1   0
3317733 1   null    2013-04-02 09:51:16.0   Kanunlar? Yapan Kim 1   0   0   1   0
3317734 1   null    2013-04-02 09:52:08.0   Secim Gecim 1   1   0   1   0
3317735 1   null    2013-04-02 09:52:55.0   Köylü Kentli    1   1   0   1   0
3317736 1   null    2013-04-02 09:53:39.0   Kanun Nam?na    1   1   0   1   0
3317737 1   null    2013-04-02 09:55:10.0   Sahipsiz    1   0   0   1   0
3317738 1   null    2013-04-02 09:56:22.0   Zorunlu Bir Dolant? 1   1   0   1   0
3317739 1   null    2013-04-02 09:57:05.0   Evde Kalm?? K?z 1   1   0   1   0
3317740 1   null    2013-04-02 09:58:25.0   S?n?rdan Geci?  1   1   0   1   0
3317741 1   null    2013-04-02 09:59:23.0   Nafile Dünya    1   1   0   1   0
3346467 1   null    2013-05-15 16:30:11.0   Muhte?em Yüzy?l Jenerik 1   1   0   1   1
3346468 1   null    2013-05-15 16:30:12.0   Dönmek  1   1   0   1   1
3346469 1   null    2013-05-15 16:30:12.0   ?ktidar Oyunu (feat. Halit Ergenc)  1   1   0   1   1
3346470 1   null    2013-05-15 16:30:13.0   Ceng-i Ali  1   1   0   1   1
3346471 1   null    2013-05-15 16:30:13.0   Sava?   1   1   0   1   1
3346472 1   null    2013-05-15 16:30:14.0   Luli (feat. Meryem Uzerli)  1   1   0   1   1
3346473 1   null    2013-05-15 16:30:14.0   Saltanat Dedikleri (feat. Bekir Ünlüataer)  1   1   0   1   1
3346474 1   null    2013-05-15 16:30:15.0   Hatice'nin A?k? 1   1   0   1   1
3346475 1   null    2013-05-15 16:30:15.0   Bismi?ah    1   1   0   1   1
3346476 1   null    2013-05-15 16:30:15.0   Ayn? Gö?ün Alt?nda  1   1   0   1   1
3346477 1   null    2013-05-15 16:30:16.0   Mahidevran  1   1   0   1   1
3346478 1   null    2013-05-15 16:30:16.0   Mohac (feat. Aytekin Ata?)  1   1   0   1   1
3346479 1   null    2013-05-15 16:30:17.0   Parga'dan Beri  1   1   0   1   1
3346480 1   null    2013-05-15 16:30:17.0   Sarayda Entrika (feat. Meryem Uzerli)   1   1   0   1   1
3346481 1   null    2013-05-15 16:30:18.0   Evvel Zaman 1   1   0   1   1
3346482 1   null    2013-05-15 16:30:18.0   Ak?nc?n?n A?k?  1   1   0   1   1
3346483 1   null    2013-05-15 16:30:19.0   A?k-? Derûn 1   1   0   1   1
3346484 1   null    2013-05-15 16:30:19.0   Nenni Desem Uyurm'ola (feat. Selma Ergec)   1   1   0   1   1
3346485 1   null    2013-05-15 16:30:19.0   ?hanet  1   1   0   1   1
3346486 1   null    2013-05-15 16:30:20.0   Yirmialt? Saat  1   1   0   1   1
3346487 1   null    2013-05-15 16:30:20.0   Hasbahcenin Gülü    1   1   0   1   1
3346488 1   null    2013-05-15 16:30:21.0   Hanedan?n Dü?ü?ü    1   1   0   1   1
3346489 1   null    2013-05-15 16:30:21.0   Ayin    1   1   0   1   1
3346490 1   null    2013-05-15 16:30:22.0   Zahir Bât?n 1   1   0   1   1
3346491 1   null    2013-05-15 16:30:22.0   Pirlere Niyaz Ederiz (feat. Erkan Bekta?)   1   1   0   1   1
```

Figure 1: Songs with null genre information

2. Neighborhood-Based (or Memory-Based) Filtering

We also did some research on recommendation methods, and gave some reports on our researches. As we stated in our project proposal, neighbor-based filtering is a kind of collaborative filtering used in recommendation algorithms. In this method, to make a prediction, memory-based collaborative filtering algorithms use the whole or a sample of the user-item database. Users are grouped as people with similar interests. In these algorithms, identifying the neighbors of a new user is needed. Neighbors are the users who have the maximum number of similar interests with new user. After identification of neighbors, a prediction of recommendation for user can be generated. [1]

The neighborhood-based collaborative filtering algorithm is the most known memory-based collaborative filtering algorithm. This algorithm uses the following steps:

I. Calculating the similarity or weight between two users or two items is the first step and it is very important in memory based collaborative filtering algorithms. There are different methods to calculate the similarity. For example, in correlation-based similarity method the formula is given below.

$$w_{u,v} = \frac{\sum_{i \in I}\left(r_{u,i} - \bar{r}_u\right)\left(r_{v,i} - \bar{r}_v\right)}{\sqrt{\sum_{i \in I}\left(r_{u,i} - \bar{r}_u\right)^2}\sqrt{\sum_{i \in I}\left(r_{v,i} - \bar{r}_v\right)^2}}$$

Here $w_{u,v}$ is the similarity between two users u and v. On the other hand, it can be $w_{i,j}$ for the similarity between two items i and j. This method is also called Pearson correlation. $i \in I$ summations are over the items that both users u and v have rated. $\bar{r}_u$ is the average rating of commonly rated items of user u. Item based similarity calculation is similar to user based one. [1]

II. Producing a prediction for the active user. In the neighborhood-based collaborative filtering algorithm, weighed aggregate of the subset of nearest neighbors of the active user is used to produce a prediction.

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u \in U}\left(r_{u,i} - \bar{r}_u\right) . w_{a,u}}{\sum_{u \in U} |w_{a,u}|}$$

Here, $\bar{r}_a$ and $\bar{r}_u$ are the average ratings of user a and user u respectively. $w_{a,u}$ is the similarity between user a and user u. $u \in U$ summations are over the users who have rated on item i. [1]

III. Generating top-N recommendations is step is to recommend N top-ranked items. There are user-based and item-based top-N recommendation algorithms. [1]

3. Evaluation Module

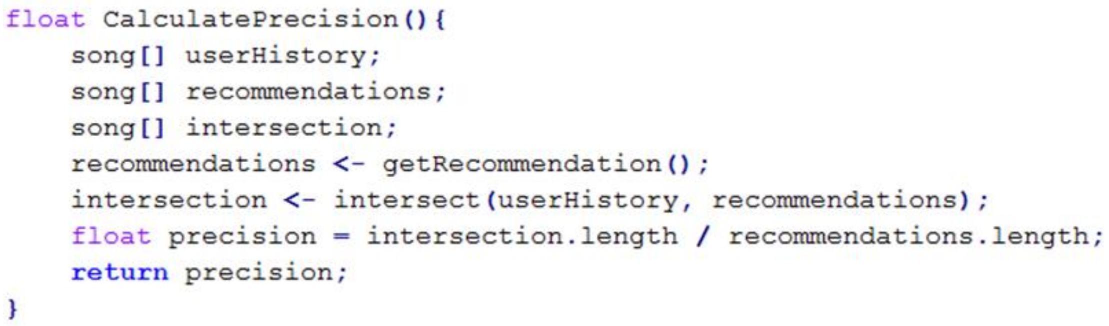
To calculate the accuracy of our recommendation algorithm, in light of our advisor’s suggestion, we decided to use one of the most known metrics for recommendation engine evaluation: “precision metric”. [2]

$$Precision = \frac{TP}{TP + FP}$$

Here in this formula, TP is the true positive value, which refers to the number of songs the user listened from the recommendation list of us. In below, the false positive value is, contrarily, the number of songs which the user has not listened from the recommendation list we provide to him/her.

In other words, we calculate the efficiency of our recommendation with looking at the ratio of the songs according to their listening states.

By referring the formula provided, we’ve acquired the below pseudo code:



```
float CalculatePrecision(){
    song[] userHistory;
    song[] recommendations;
    song[] intersection;
    recommendations <- getRecommendation();
    intersection <- intersect(userHistory, recommendations);
    float precision = intersection.length / recommendations.length;
    return precision;
}
```

Figure 2: Pseudo code of precision calculation for evaluation module

Updated Gantt Chart for Our Future Works:

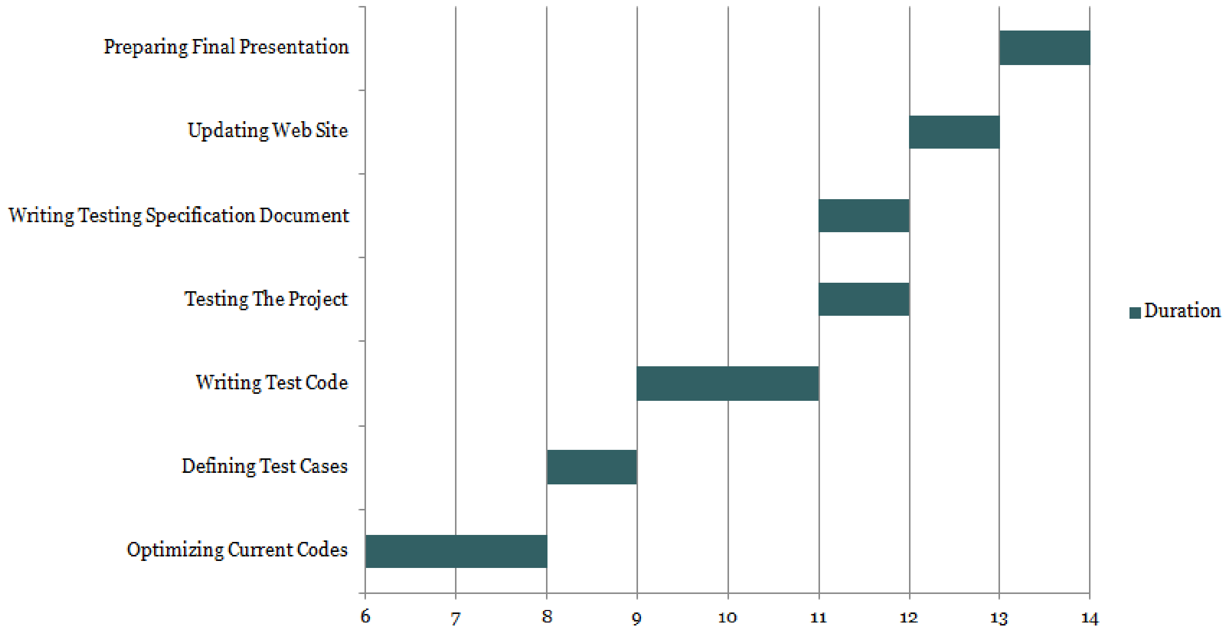


Figure 3: Gantt Chart of our future works

Finishing Evaluation Module:

We will finish implementing our Evaluation Module.

Solving the Inaccurate Rating Information In the DB:

The rating values that are in the logs result in inaccurate calculations during the recommendation. For example it is very common in the logs that some users have the same rating value for all of the songs they have listened. Since this result in mathematically impossible calculations, we will either try to improve the rating information in the DB, or alter our algorithm to prevent erroneous calculations.

Final Improvements on Collaborative Filtering:

We will do our final touch on the recommendation algorithm and try to optimize our code for faster computation.

Offline Similarity Calculations:

Since our neighborhood-based collaborative filtering needs frequent user similarity values during computation, the system calculates hundreds, sometimes thousands of user similarity values. We will solve this by calculating the user similarity values between users offline and keep them in the database, so that the online part of the system will perform the tasks a lot faster.

Optimizing DB structure:

Since our recommendation and evaluation algorithms will be finalized in this iteration, we may update our database structure for a more efficient system.

References:

[1] Hong, Jun.(August 2009). A Survey of Collaborative Filtering Techniques. Advances in Artificial Inteligence. Retrieved from: http://www.hindawi.com/journals/aai/2009/421425/

[2] Ricci, Francesco. Rokach, Lior. Shapira, Brancha. Kantor, Paul. (2010). *Recommender Systems Handbook.* New York.