Analysis and Evaluation of Recommendation Systems

Emiko Orimo[1,2], Hideki Koike[2], Toshiyuki Masui[3], and Akikazu Takeuchi[1]

[1] So-net Entertainment Corporation
[2] University of Electro-Communications
[3] Apple Computer Inc.
{emiko.orimo,akikazu.takeuchi}@so-net.co.jp, koike@is.uec.ac.jp, masui@pitecan.com

Abstract. Popular online services, such as Amazon.com, provide recommendations for users by using other users' rating scores for items. In this study, we describe three types of rating systems: score-rated, count-rated, and digital-rated. We hypothesize that digital-rated systems provide the most useful recommendations. Then we analyze the differences in the results of the rating when the granularity of the score changes. Finally, we visualize users by developing a 2-D visualization system that uses a multi-dimensional scaling method.

Keywords: recommendation system, rating algorithm, multi-dimensional scaling method, visualization.

1 Introduction

Popular online services, such as Amazon.com, provide a recommendation to users when a user opens a page describing an item or clicks a link to view its detail. Finding items from the recommendation list is accomplished by information retrieval using methods such as keyword search or browsing.

This recommendation method works well when the user has no exact target, but it lacks quantitative value. In general, calculation of recommended items utilizes a user's history of purchases or rating scores for items. For example, the user gives a score between 1 and 10 based on his or her evaluation of the item. Such rating scores, however, are not exact because people tend to give high scores such as 9 or 10. Ratings can be recognized as “interest in items.” Thus, it might be thought that an item rated as a 9 or 10 by a user means that it is his or her favorite, but the rating is not definitive. Therefore, a binary rating such as “buy or not” or “listen or not” could provide a more useful recommendation. We hypothesize that such a binary rating makes it easier for users to rate items and also makes it easier for recommendation systems to perform calculations.

In this paper, we first observe existing rating systems. Then we analyze the difference between the results when the granularity of the rating scores changes. In order to analyze the results visually, we developed a 2-D visualization system that visualizes users who are making recommendations using a multi-dimensional scaling (MDS) method. MDS is widely used in various fields to analyze mutual relations among items. The quantification theory type III (QT-III) enables calculation of the

M.J. Smith, G. Salvendy (Eds.): Human Interface, Part I, HCII 2007, LNCS 4557, pp. 144–152, 2007.

"distance" between items. Using this distance, we can decide the geometrical position of each item so that similar items are placed physically near each other. For example, if user A answered that "Oasis" and "Beatles" are his or her favorite artists, "Oasis" and "Beatles" are near each other with respect to user A. In the same way, if user A and user B answered that "Oasis" is their favorite artist, users A and B are near each other with respect to "Oasis".

2 A Study of Rating Methods

In this study, we consider three types of rating methods for items. The first method gives regulated rating scores such as five stars. We call this a "score-rated type". The next method counts a user's actions such as history of purchase. We call this a "count-rated type". The third method expresses a user's interest in terms such as "1 or 0", meaning "I like it" or "I don't like it." We call this a "digital-rated type". In the following sections of this paper, we analyze the differences between the results obtained by each rating type.

2.1 Samples of Each Type of Rating

Score-Rated Type: Ratebeer.com. Ratebeer.com is a web service about beer. It has a huge amount of information about beer and also rating data by its users.

Fig. 1. Ratebeer.com

Once a user gives ratings about aroma, appearance, flavor, palate, and overall impression, Ratebeer.com converts them to official scores between 0.0 and 5.0. In this case, Ratebeer.com is categorized as a score-rated type.

Count-Rated Type : Last.fm. Last.fm is a web service related to music. This service stores users' histories of listening in real time. Using these histories, it provides recommended tracks and artists to each user. Last.fm is categorized as a count-rated type.

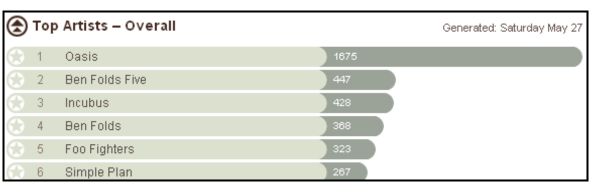

Fig. 2. Last.fm

Digital-Rated Type : Hondana.org. Hondana.org is an online bookshelf service. Its users can register any books they have. It does not require the users to rate books. Hondana.org can be categorized as a digital-rated type.

Fig. 3. Hondana.org

2.2 Creating a Data Set

Currently, there are many web services such as Flickr or del.icio.us that feed XML documents like RSS or Atom. In order to collect as much real rating data as easily as possible, we used the XML feeds and created ratings data from them.

Getting data from Web service. Service providers gather and use users' data for their own purposes. They also deliver the information as an XML document called an RSS feed. We can utilize this XML document in our applications.

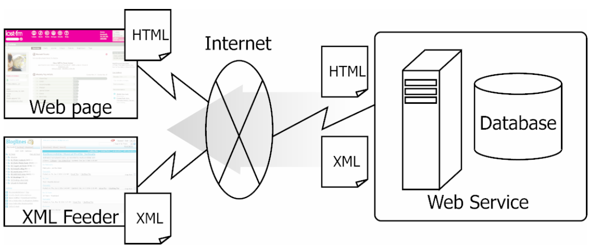

Fig. 4. Web service

Last.fm and Audioscrobbler.net. Audioscrobbler.net provides XML documents. Its data source is the users' listening habits at Last.fm. Using Audioscrobbler, it is possible to get such data as a profile, top artists, top albums, and top tracks for each user.

Fig. 5. Web service of Audioscrobbler

Figure 6 shows the XML about top tracks.


```
<topartists user="orimo">
<artist>
<name>Oasis</name>
<mbid/>
<playcount>1768</playcount>
<rank>1</rank>
<url>http://www.last.fm/music/Oasis</url>
</artist>
<artist>
<name>Ben Folds</name>
<mbid/>
<playcount>702</playcount>
<rank>2</rank>
<url>http://www.last.fm/music/Ben+Folds</url>
</artist>
                    -snip-
</topartists>
```


Fig. 6. Users' Top Artists XML

3 Study of Each Data Type

We obtained top favorite artist data for 100 users from Audioscrobbler and conducted experiments described in the following sections. We converted these data to category data for the QT-III, which we called the original data set.

3.1 Original Data Set

In the original data set, it is very rare to see that two or more users listen to the same artist the same number of times. Consider, for example, user A who listened to a track by Oasis, for instance, 10 times and user B who listened to the same track 200 times.

Original Data

Track Data

track	orimo	koike	mifo
Oasis	429	0	40
LedZeppelin	0	598	20
Blur	18	22	23

Category Data

track	orimo	koike	mifo
Oasis40	0	0	1
Oasis429	1	0	0
LedZeppelin20	0	0	1
LedZeppelin598	0	1	0
Blur18	1	0	0
Blur22	0	1	0
Blur23	0	0	1

Normalized Data

Track Data

track	orimo	koike	mifo
Oasis	100	0	9
LedZeppelin	0	100	3
Blur	78	95	100

Category Data

track	orimo	koike	mifo
Oasis9	0	0	1
Oasis100	1	0	0
LedZeppelin3	0	0	1
LedZeppelin100	0	1	0
Blur88	1	0	0
Blur95	0	1	0
Blur100	0	0	1

Fig. 7. Original Data Set and Normalized Data Set

If A listened to a particular song 10 times and B listened to the same song 10 times, the result might be given as “A's listening habit is similar to that of B” since both listened to the song the same number of times. This is not appropriate because the total number of A's playing count is different from that of B.

It is, therefore, necessary to normalize the data. By setting max repeat count to 100, we normalized all repeat counts. For example, if user A listens to a song by Oasis 10 times and a song by Blur 5 times, the value of Oasis is set to 100 and that of Blur is set to 50. Using this normalized data, we calculate similarity of users’ listening by QT-III.

3.2 Normalized Data Set

We converted the original data into a normalized data set, as shown in fig.7. Then we analyzed the difference in the result of the rating when the granularity of the score changed. In order to compare the results visually, we developed a 2-D visualization system that categorizes items by using MDS. Using this data set, we created a temporal data set of the count-rated type. Then we observed how users' positions changed on 2-D space.

3.3 Score-Rated Data Set

To create a temporal data set of the count-rated type, we converted the normalized data set as shown in fig.8. We call this result a score-rated data set. In ten grades of scoring, the user can give a score from 1 to 10. Since such fine granularity makes the rating complicated, few services use this rating. A system using five grades of scoring, as seen in YouTube.com, is more popular. In this case, users' positions are calculated and are plotted on a 2-D map as shown in fig.9. Both rating methods, however, have the problem that there is no exact rule for scoring, and this might reduce the reliability of the rating. In two grades of scoring, on the other hand, the user chooses “good” or “not good”. In this case, users' position are calculated and plotted as shown in fig.10.

Scoring in Ten Grades

track	orimo	koike	mifo
Oasis1	0	0	1
Oasis10	1	0	0
LedZeppelin1	0	0	1
LedZeppelin10	0	1	0
Blur8	1	0	0
Blur9	0	1	0
Blur10	0	0	1

Scoring in Five Grades

track	orimo	koike	mifo
Oasis1	0	0	1
Oasis5	1	0	0
LedZeppelin1	0	0	1
LedZeppelin5	0	1	0
Blur4	1	0	0
Blur5	0	1	1

Scoring in Three Grades

track	orimo	koike	mifo
Oasis1	0	0	1
Oasis3	1	0	0
LedZeppelin1	0	0	1
LedZeppelin3	0	1	0
Blur3	1	1	1

Scoring in Two Grades

track	orimo	koike	mifo
Oasis1	0	0	1
Oasis2	1	0	0
LedZeppelin1	0	0	1
LedZeppelin2	0	1	0
Blur2	1	1	1

Fig. 8. Score-Rated Data Sets

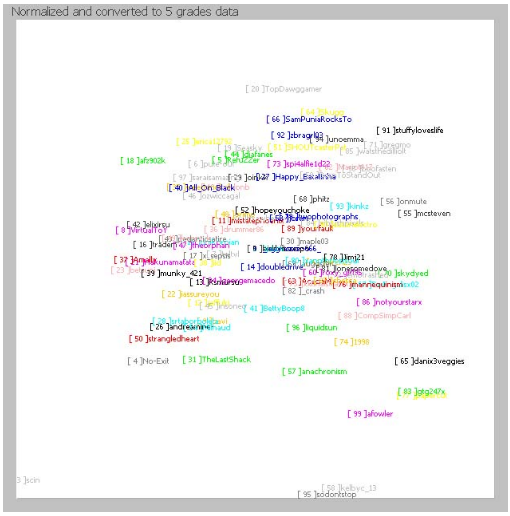

Fig. 9. Users' map using Scoring Data Set

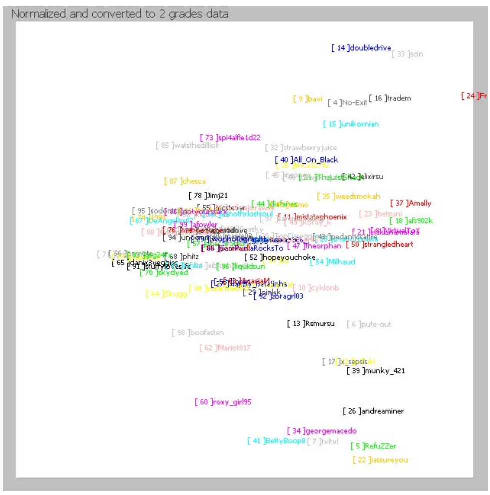

Fig. 10. User map using Scoring Two Grades data Set

3.4 Digital-Rated Data Set

In the next step of this study, we converted the original data set to a digital-rated data set. In this data set, we consider that "repeating artist A's track 100 times" and "repeating artist A's track 1 time" are the same. Scoring in digital is a rating based on selecting "yes" or "don't care". In this case, users' positions are calculated and plotted as shown in fig.11.

3.5 Study of Each Data Set

Using coordinate values calculated by QT-III, we drew a line graph (fig. 12) to compare changes of results from each score-rated data set. This shows that the ups and downs of the graph are almost synchronized, indicating that the granularity of the rating is not very important.

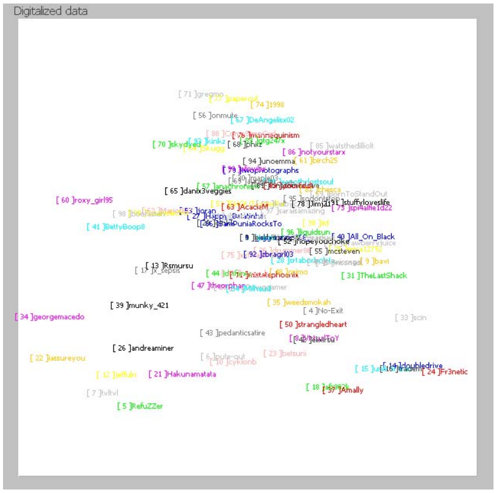

Fig. 11. User map using Digital-rated data Set

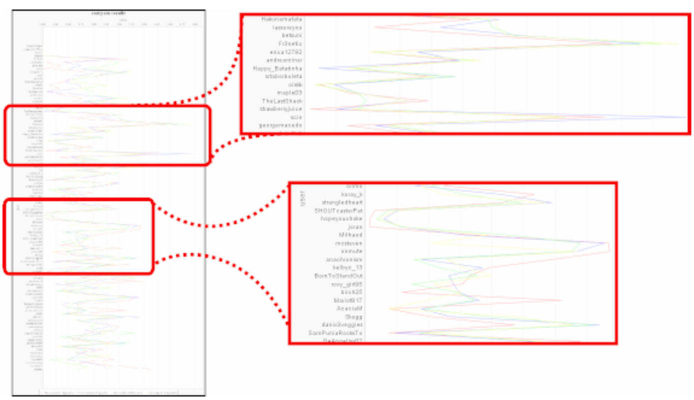

Fig. 12. Comparing results of Score-Rated data sets

Next, we compared the 2D maps generated by two-grade scoring and scoring in digital-rated scoring. The user “orimo” was set at the origin of coordinates and other users were placed by using the distances from “orimo” (fig. 13).

Digitalized Data

1	srtaborboleta	0.155149316
2	drummer86	0.346965388
3	weedsmokah	0.403826381
4	zbragr103	0.411075586
5	hopeyouchoke	0.41732734
6	emotrashed	0.44618187
7	Seasky	0.508617901
8	ozwiccagal	0.509199323
9	mistatephoenix	0.523423148
10	liquidsun	0.538828165
11	mcsteven	0.589467943
12	Opium	0.59485573
13	_aggressor-666_	0.59485573
14	anabiacrespo	0.59485573
15	bizkvit	0.59485573
16	Milhaud	0.608140413
17	insoneo	0.609538465
18	koray_k	0.624152322
19	xxfinalstrawxx	0.661727439
20	No-Exit	0.673487702
21	strangledheart	0.69949417
22	TopDawggamer	0.75158572
23	iid	0.774654069
24	strawberryjuice	0.806508199
25	saraisamazing	0.808991997

Scoring in Two Grades

1	mistatephoenix	0.191449835
2	saraisamazing	0.222729065
3	diafanes	0.243777247
4	koray_k	0.293782116
5	insoneo	0.387704894
6	ozwiccagal	0.388227254
7	TopDawggamer	0.455777215
8	drummer86	0.480567861
9	TheLastShack	0.518400881
10	erica12792	0.523523695
11	theorphan	0.583546913
12	All_On_Black	0.590980357
13	weedsmokah	0.624140583
14	Opium	0.637625105
15	_aggressor-666_	0.637625105
16	anabiacrespo	0.637625105
17	bizkvit	0.637625105
18	pedanticsatire	0.692259599
19	Seasky	0.724105116
20	srtaborboleta	0.737082948
21	iid	0.739974256
22	strawberryjuice	0.747340238
23	hopeyouchoke	0.771995204
24	xxfinalstrawxx	0.7999411
25	xanothrlostsoul	0.897813515

Fig. 13. Distances to Origin Point User

Since the top 25 similar users are almost the same, it could be said that both data sets could provide almost the same result. We also analyzed the reason why the distances between these users and "orimo" are close. Most similar users in the digitized data set were almost repeating the same artist's track. Most similar users in the two-grade scoring data set were repeating tracks of an artist similar to orimo's favorite. So it seems that both data sets apply to similar users.

4 Visualization System

We developed a visualization system using Apache Tomcat on Windows XP. The system was implemented using a Java Servlet and Java applets. We acquired XML

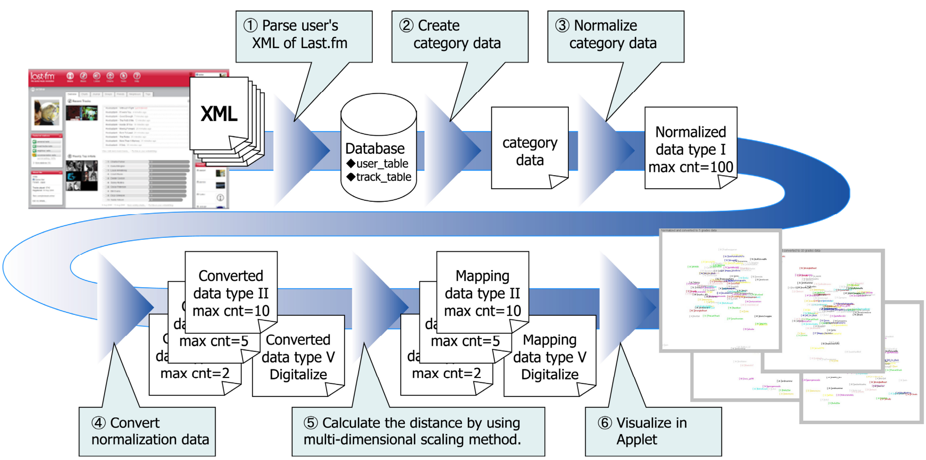

Fig. 14. Process of Calculating Mapping Data

data of Last.fm via HTTP. Using a Java Servlet, the original data are converted to normalized data. Finally, each user is visualized by using a Java applet. This process is shown in fig.14.

5 Related Work

The authors previously developed a visual browsing system for a movie database. The system, named ZASH, visualized movies, recommenders, actors, directors, and keywords on different 2-D planes in one 3-D space. Movies and recommenders are categorized by using MDS QT-III, and therefore similar movies are displayed physically near each other. One of the problems in ZASH is it requires users to give scores to the movies.

TechLens+ is a hybrid recommender algorithm that combines collaborative filtering and content-based filtering to recommend research papers to users. Through some experiments, it is shown that the algorithm gives a good recommendation. However, it also requires the users to give scores to papers.

Amazon.com uses recommendation algorithms to personalize the online store for each customer. The available selection radically changes based on customer interests. Amazon uses an algorithm called item-to-item collaborative filtering, but it also requires users to rate items.

6 Conclusion and Future Works

In this paper, we observed existing rating methods and identified that the granularity of the rating scores is not very important in calculating the similarities of users. In order to verify our hypothesis, we developed a system that collects a large data set from the Internet, normalizes the data, calculates the similarities of users by using MDS QT-III, and visualizes them on a 2D map. The experimental results support our hypothesis.

As a future project, we will collect much more data and analyze the similarities of users. Then we want to apply our method to the recommendation system.

References

1. Orimo, E., Koike, H.: ZASH: A Browsing System for Multi-Dimensional Data. In: Proc.1999 IEEE/CS Symposium on Visual Languages (VL'99), pp. 288–295 (1999)
2. Torres, R., et al.: Enhancing Digital Libraries with TechLens+. In: Proc.2004 Joint ACM/IEEE Conference on Digital Libraries(JCDL'04), pp. 228–236 (2004)
3. Greg, L., Brent, S., Jeremy, Y.: Amazon.com Recommendations. Item-to-Item Collaborative Filtering. IEEE Internet Computing, pp. 76–80 (January/February 2003)
4. Tatemura, J.: Visualizing Document Space by Force-directed Dynamic Layout. In: Proc. 1997 IEEE Symposium on Visual Languages (VL'97), pp. 119–120 (1997)
5. Maneeroj, S., et al.: Combining Dynamic Agents and Collaborative Filtering without Sparsity Rating Problem for Better Recommendation Quality. DELOS Workshop: Personalization and Recommender Systems in Digital Libraries (2001)