Empirical Study of User Preferences Based on Rating Data of Movies

YingSi Zhao1*, Bo Shen2

1 School of Economics and Management, Beijing Jiaotong University, Beijing, 100044, China, 2 School of Electronic and Information Engineering, Key Laboratory of Communication and Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing, 100044, China

* yszhao@bjtu.edu.cn

Abstract

User preference plays a prominent role in many fields, including electronic commerce, social opinion, and Internet search engines. Particularly in recommender systems, it directly influences the accuracy of the recommendation. Though many methods have been presented, most of these have only focused on how to improve the recommendation results. In this paper, we introduce an empirical study of user preferences based on a set of rating data about movies. We develop a simple statistical method to investigate the characteristics of user preferences. We find that the movies have potential characteristics of closure, which results in the formation of numerous cliques with a power-law size distribution. We also find that a user related to a small clique always has similar opinions on the movies in this clique. Then, we suggest a user preference model, which can eliminate the predictions that are considered to be impracticable. Numerical results show that the model can reflect user preference with remarkable accuracy when data elimination is allowed, and random factors in the rating data make prediction error inevitable. In further research, we will investigate many other rating data sets to examine the universality of our findings.

Introduction

User preferences are considered to be the user’s opinions on social topics, goods, services, friends, works, ads, the search results of search engines, and more. Ordinarily, user preferences are closely related to recommender systems, because the task of a recommender system is to convert data on users and their preferences into predictions of their possible interests [1, 2]. Although recommender systems are not the only places to display the prowess of user preferences, they generate a heavy demand for user preferences, and create huge amounts of data, which provides the opportunity to mine and learn more characteristics of the user preferences.

In the recommender system field, researchers mainly focus on how to improve the accuracy of recommendations [3–8], which implicitly involves how to obtain user preferences. One important method is collaborative filtering (CF) [9]. CF is based on the fact that people make their decisions about new things based on their own knowledge history, as well as the experiences of other related people [2], e.g., as expressed on Amazon’s website: “Customers Who Bought This Item Also Bought.” Of course, in a recommender system, CF is considered to be a
kind of data filtering algorithm. In CF models, the key issue is how to measure the similarity between users [10–12] or between items [3, 12, 13], which directly concerns the degree of correlation between the analyzed target and other reference objects. The common methods of similarity measurement include overlap [14], Euclidean distance [15], Hamming distance [16], Pearson correlation [17], and the cosine of the angle between vectors [18]. There are also many improved and adjusted methods [1, 2, 18] based on traditional metrics in the literature. Each of these methods has its own advantage, and no method wins out over all others. However, it is commonly recognized that the similarity between items tends to be more static than the similarity between users [1]. Model-based methods are also available, which include SVD [19], LSA [20], Bayesian [21], fuzzy [22], and neural networks [23]. These methods aim to directly calculate recommendations through pre-created models rather than by obtaining the relation between users or items. Thus, user preferences are hidden in models. Some of these models are also used to reduce the dimensionality of the data, such as SVD and LSA. In addition, they usually have higher commendation accuracy. Because of the absence of explicit physical meanings about user preferences, it is usually difficult to improve these methods and to understand how users make decisions by them.

Whether CF or model-based methods are used, history data about users and items are the basis. There are two kinds of data: two-valued data and multiple-valued data. Two-valued data only convey “like” and “dislike” opinions from users about an item. Multiple-valued data contain the ratings of users for items, which in general are integers with a range of 1–5. Ratings can be regarded as a kind of reflection of user preferences on the dimension of a certain object. For simplicity, some researchers map multiple-valued data to two-valued data, especially when the purpose of the study is to find general rules about user preferences [14, 24].

Although more and more factors are being included in recommender systems, and new algorithms are continually presented, what affects a user’s decisions and whether it can be predicted accurately are still open issues. In particular scenarios, other researchers attempt to determine the key factors that affect user preferences. Ref [24] presented a weighting method to extracting the hidden information of networks formed by users and items. By assigning a heterogeneous distribution of initial resources [16] and removing the redundant correlations [25], the original method and its improved methods find several factors related to user preferences. In Ref [14], statistical methods were used to explore affinity relations. The authors found that there was an intrinsic limit, which would prevent the achievement of perfect prediction by statistical means, even if more data were obtained.

From another viewpoint, the development of computational social science makes it possible to study human behavior using online data [26]. The authors of Ref. [27] found that, through analyzing massive data, a better understanding of collective human behavior could be achieved, and more evidences were presented in [28]. Further, the research of [29] indicated that users’ collective future behavior can be predicted by what they searched for online. The research on social opinion has some goals in common with recommender systems. For example, researchers want to know how people format or change their opinion about a given topic [30]. Many theories have been developed, such as the majority rule model [31], social impact theory [32], and bounded confidence model [33], which could also be considered to be methods for understanding user preferences. However, unlike the study of social opinion, recommender systems do not consider the macroscopic state and the evolution process of user preferences.

In this paper, we introduce the results of an empirical study of user preferences based on rating data. We first analyze the relationships between users and items, and then map them into a hyper-network. We present a kind of distance measure method, and find some interesting characteristics about user preferences. Based on our findings, we propose a user preference model, which employs the relations between items and a user’s history ratings to evaluate their
preference for new items. We also discuss the results of the proposed model. It should be noted that in this paper, we only take into account the empirical study of a user’s preference using a special data set, instead of building a recommendation algorithm or a recommender system.

### Empirical Analysis

In this paper, we use one of the standard benchmark data sets, namely MovieLens [34], to carry out our analysis. The data set we used contains 100,000 ratings by 943 users on 1,682 movies. Each rating item is an integer in the range of 1–5. In one example, listed in Table 1, users u_1, u_2, u_3, and u_4 provide ratings for movies a, b, and c. These ratings can be regarded as a kind of relationship between the users and movies.

The relationship can also be presented in the form of a network, as shown in Fig 1. From the viewpoint of the network, the nodes of movies connect users together, and the nodes of users connect movies together. Obviously, there are two kinds of different nodes in these networks, user nodes and movie nodes. If the movie nodes are extracted, the network will have the structure shown in Fig 2, which is a so-called hyper-network [35–37].

A hyper-network consists of pairs $H = (V, E)$, where $V = \{v_1, v_2, \ldots, v_n\}$ is the set of nodes, and $E = \{e_1, e_2, \ldots, e_m\}$ is the set of hyper-edges, with $e_i \subseteq V$ for $i = 1, 2, \ldots, m$ [37]. Clearly, in a hyper-network, each hyper-edge is a subset of the set of nodes and contains at least two nodes, as illustrated in Fig 2. Here, each movie is a hyper-edge, e.g., movie a is associated with users

| Table 1. Relationship between users and movies based on ratings. |
|------------------|---|---|
|                  | a | b |
| u_1              | $r_{1a}$ | $\varnothing$ | $\varnothing$ |
| u_2              | $r_{2a}$ | $r_{2b}$ | $\varnothing$ |
| u_3              | $\varnothing$ | $r_{3b}$ | $r_{3c}$ |
| u_4              | $r_{4a}$ | $\varnothing$ | $r_{4c}$ |

doi:10.1371/journal.pone.0146541.t001

doi:10.1371/journal.pone.0146541.g001
{u₁, u₂, u₄}, b with {u₂, u₃}, and c with {u₃, u₄}. In addition, we can deem that the nodes belonging to a hyper-edge fully connect to each other.

From the perspective of the hyper-edge, a hyper-network can be defined as a set of \( R \), which is the relation between two sets \( A \) and \( B \) [35]:

\[
R(a) = \{ b \mid b \in B, a \rightarrow b \}
\]

(1)

\[
H_a(R, B) = \{ R(a) \mid a \in A \}
\]

(2)

Here, \( a \rightarrow b \) means distinctly that \( a \) relates to \( b \). Let all movies be set \( A \). Let all users be set \( B \), and let all ratings be the relation between \( A \) and \( B \). Then, the network in Fig 2 can just be mapped into a bipartite hyper-network, as shown in Fig 3. In this bipartite hyper-network, movies correspond to hyper-edges, e.g., movie \( a \) corresponds to hyper-edge \( e_a \), i.e., \( R(a) \), which is a subset of user set \( B \). A bipartite network can be used to describe many-to-many relations with two object sets in the real world, such as a flavor network [38], scientific collaboration network [39], users and products network [24] and so on. Many researchers employ bipartite networks as a tool to study relations [40].

From the hyper-edge viewpoint, the estimation of how a user will like a movie can be converted, to some extent, into finding what correlation exists between the hyper-edges that the user belongs to and the hyper-edge that the user will belong to. For example, if we need to predict the opinion of user \( u_3 \) in Fig 3 about movie \( a \), the correlations between hyper-edge \( e_b \) and \( e_a \), and between \( e_c \) and \( e_a \), may provide useful information.

The hyper-edge characteristics can usually be modeled by employing the concept of the simplex volume because a hyper-edge is regarded as a simplex [41]. However, simplex volumes
degenerate when the degree of the hyper-edge is larger than the dimensions of the feature \([42, 43]\), which is exactly the case when treating movies as hyper-edges.

We define the distance between two hyper-edges \(e_i\) and \(e_j\) as follows.

\[
d_{ij} = \sqrt{\phi_{ij} \sum_{k \in R(i) \cap R(j)} \left( \frac{r_{ik} - r_{jk}}{s_k} \right)^2}
\]

\[
a_{ij} = \frac{|R(i) \cup R(j)|}{|R(i) \cap R(j)|}^{\frac{1}{2}}
\]

\[
\phi_{ij} = \frac{|R(i) \cap R(j)|}{|R(i) \cap R(j)| + 1}
\]

where \(r_{xy}\) is the rating that user \(x\) gave to movie \(y\), \(s_k\) is the standard deviation between \(r_{ik}\) and \(r_{jk}\), and \(|X|\) denotes the number of elements in hyper-edge \(X\).

\(\phi_{ij}\) is called the shrinking factor and is used to eliminate the cumulative effect of the standardized difference between two ratings. \(a_{ij}\), called the stretching factor, is designed to reflect the extent that the two hyper-edges overlap their union. Obviously, when \(R(i) = R(j)\), we have \(a_{ij} = 1\), and if \(R(i) \cap R(j) = \emptyset\), then \(a_{ij} = +\infty\). This seems a reasonable measurement of the correlation between movies represented by hyper-edges. When more users gave them the same ratings, the more common characteristics they could have.

We calculated the distance between any two movies in the MovieLens data set using Eq 3. The distance data were stored in S1 File. Fig 4 plots the network of data set u.data by Crytoscape [44]. In the plot, each node represents one movie, and a movie only has one connection to
its first-order nearest neighbor (S2 File) in the sense of distance defined by Eq 3. For simplicity, we call the nearest neighbor the h-neighbor and the connecting relation h-connected.
The results show that these movies form many sub-networks (named cliques here) with different sizes, and there is no connection between these cliques (286 cliques for data set u.data). This implies that the movies in the dataset have the potential characteristic of closure, which could be the result of users’ selections with explicit preferences. Then, the closure feature of the cliques could be used to evaluate the preferences of users who have voted for some of the movies in a clique. Similar clique structure also appears on other networks, such as Flickr and CiteUlike [45].

Furthermore, we also notice that only a small number of cliques contain a large number of nodes, while most have only a few nodes under the condition of first-order h-neighbors. We plot the statistical results in Fig 5, which shows that the distribution of the clique sizes closely follows a power law: $S(x) \sim x^{-\tau}$, where $\tau$ is a constant exponent with a value of about 1.65. Similar phenomenon was also observed in many other real systems which can be modeled as bipartite networks [1]. For example, the item-degree distributions of the e-commerce data in amazon.com [46], the music sharing data in audioscrobbler.com [47] and the movie data in the Internet Movie Database [48] all obey power-law-like form with different exponent value.

One possible explanation for this power law is that the numbers of ratings received by the movies are inhomogeneous. Because the data in the movie data set was collected during a short period of seven months, new movies at that time evidently received more ratings, while old movies got less attention. Although every user gave at least 20 ratings, about 44.8% of the movies had less than 20 rating, and about 79% of the movies had less than 94 ratings, which is 1/10 of the number of users.

Fig 5. The size of cliques vs. the number of corresponding networks.

doi:10.1371/journal.pone.0146541.g005
When connecting movies by first-order h-neighbor, those movies that had fewer ratings would choose the h-neighbor in a greater range. Thus, more nodes connected together, and few large networks formed. We would expect the size of the cliques to become more homogeneous when data could be retrieved over a long time range. However, even in the data for a prolonged time period, the differences between users and the differences between movies will still lead to various cliques.

We also notice that there are many pairs of nodes, which are the first-order h-neighbors of each other, and each clique has one pair of such nodes, expressed as \( \Lambda(1) = \{(\alpha, \beta)|\alpha \sim \beta, \alpha, \beta \in H_A\} \). If a clique has only two nodes and they are \( \Lambda(1) \), we call it the first-order h-neighbor clique. This indicates that some common characteristics bring them together with a stronger connection, which may cut off their relations to other nodes when the first-order h-neighbor rule is applied. For example, the nodes representing the movies *Batman Forever* (1995) and *Batman Returns* (1992) connect together to form a clique with two nodes. They are the h-neighbors of each other. Under the rule of the first-order h-neighbor, the existence of these nodes is the reason for the closure of the clique.

We found the statistics for all the ratings of the movies that belong to \( \Lambda(1) \) using the following method:

\[
P(\varepsilon) = \sum_{\kappa \in \Lambda(1)} \delta(\varepsilon_k, \varepsilon) / N,
\]

where \( \varepsilon_k \) is the normalized RMSE of the ratings users gave to a pair of nearest nodes, \( N \) is the number of the pair, and \( P(\varepsilon) \) is the distribution of the normalized RMSE. \( \delta \) is the Kronecker symbol. The results are plotted in Fig 6. Clearly, 75% of the normalized RMSE values lie in the

![Fig 6. The distribution of RMSE of ratings that users voted for the first-order h-neighbor clique.](Image)

doi:10.1371/journal.pone.0146541.g006
range of 0.05–0.35. The maximal RMSE is 4.0. Thus, the ratings that users gave to the first-order h-neighbor nodes have RMSE values of 0.2–1.4, and most are less than 1.0. This means that most users have similar opinions about \( \Lambda(1) \). That is to say, from the viewpoint of the users, these two movies show a strong similarity. It should be further emphasized that here the h-neighbor nodes forming a clique are movies. Although the first-order h-neighbor clique implies these movies have similarity and a user related to these movies has similar opinions on them, it does not mean that these users related to a clique have the same preference on different types of movies. It should also be noted that here the RMSE value is not comparable with that used in predicting precision, because it is calculated for two different movies.

We further measure the network constructed by the second-order h-neighbor rule (S3 File), in which the first-order h-neighbors are included. In this case, the closure characteristic almost disappears, as shown in Fig 7. The detailed data indicate that the diversity of the distance increases under the condition of the second-order h-neighbor rule, which causes more nodes to connect together. In other words, the second-order h-neighbors make connections between the nodes that are discrete under the first-order neighbor rule, as shown by the links indicated by the red arrows in Fig 7.

Obviously, the second-order h-neighbor rule blurs the edge of the cliques formed by the first-order h-neighbor rule. Although more nodes are connected into cliques, the similarity between the nodes in a clique is reduced. Thus, using a distance threshold may be a better idea to keep the closure and avoid the diverse distance effect. We will study this in more depth in the future.

We also investigate the distribution of the distance between movies using the method reported in ref. [14] with Eq 7. The distribution plot is shown in Fig 8.

\[
P(d) = \sum_i \sum_{j \neq i} \delta(d_{ij}, d) / N(N - 1)
\]

where \( \delta \) is the Kronecker symbol, \( N \) is the number of movies, and \( d_{ij} \) is the distance between movies \( i \) and \( j \) by Eq 3.

Clearly, the distance is a rather homogenous distribution as a result of \( P(td) = t^{-1} P(d) \). This is essentially in agreement with the result of ref. [14] which was achieved on the EachMovie data set, except that the distribution of the distances between movies does not appear to be polarized. Moreover, its peak is less than 0.2 and around \( d \sim 0.17 \), which means, according to the result of ref. [14], that we would be able to use the information contained in the relations between movies to describe user preferences and predict their ratings. It also means that we do not need the information about the similarity between users.

**User Preference Model and Results**

In the above empirical analysis of the MovieLens data, we found that the hyper-network of movies shows the characteristic of closure under the condition of considering only the first-order h-connected, and the sizes of these closure cliques demonstrate a power law distribution, which reflects the existence of some interdependency between some movies, and users’ opinions about movies have potential tendencies. The distribution of the distance between any two movies gives further evidence that the relations between movies can be used to describe the preferences of users.

The basic idea is to use the information about the relations between movies to estimate users’ opinions: if we want to know the opinion of user \( i \) about movie \( a \), we could use the opinion of user \( i \) about movie \( b \) that is a first-order h-neighbor of movie \( a \) for the estimate.

However, there are still two obvious issues to be considered:
I. Many first-order $h$-connected cliques are too small.

II. Lots of movies have fewer ratings.

For a clique with a small size, if user $i$ rated movie $a$, then predicting the rating that user $i$ will give to movie $b$ is reasonable when $a$ and $b$ are first-order $h$-neighbors. In contrast, if user $i$ did not rate any movie in a clique, the prediction for the movies in this clique will become unreasonable.

Fig 7. Cliques in which each node only connects to its second-order $h$-neighbors, plotted by Crytoscape [44].

doi:10.1371/journal.pone.0146541.g007
In consideration of the above empirical analysis results and to overcome these issues, we present a user preference model, as follows:

\[ Y_i(\beta) = \frac{\sum x r_{ix} d_{xp}}{\sum d_{xp}} \quad x \in M = \left\{ m | \arg \max_{m} (d_{xp}) \right\} \] (8)

where \( Y_i(\beta) \) denotes the estimation of the opinion of user \( i \) about movie \( \beta \), \( r_{ix} \) is the rating that user \( i \) gave movie \( x \), and \( d_{xp} \) is the distance between movies \( x \) and \( \beta \). \( \{ m | \arg \max_{m} (d_{xp}) \} \) is the set that contains the nearest \( k \) movies to movie \( \beta \), where \( k \) is a tunable parameter. \( \in \) means taking members contiguously.

Obviously, the user preference model employs more than one movie and their ratings to eliminate the influence of issue I. According to the previously mentioned analysis result, the rating of a movie with a small distance to movie \( \beta \) will have more influence on user \( i \). Thus, we introduce a weight for the rating value, based on the distance.

Fig 9 gives the results of applying the presented user preference model to MovieLens data set ua, which has a test data set ua.test with exactly 10 ratings per user. The result data are stored in S4 File, S5 File, S6 File, S7 File and S8 File. To compare it with other typical recommendation algorithms, we use the RMSE as the evaluating indicator of the prediction accuracy.

The red curve in Fig 9 shows that the \( k \leq 2 \) prediction error is large, which reflects a difference of opinion between users when they face similar movies. After that the ratings of more similar movies are taken into account, the prediction error of the presented model decreases. The green dashed lines in Fig 9 are the best RMSE values of four typical algorithms [49–52] used for the same data set [1]. When \( k \geq 5 \), the presented model can obtain a smaller error of \( RMSE \leq 0.8447 \).
Increasing $k$ means more data are used to predict the user’s opinion. By common sense, this will continuously enhance the prediction accuracy. However, the result in Fig 9 implies that there is a limit. When $k$ is small, adding data is helpful for increasing the prediction accuracy. When $k > 9$, for data set ua, the prediction error begins to increase. This agrees with the phenomenon of the saturation of the prediction power mentioned in ref. [14]. We also checked other data sets of MovieLens, including u1~u5, which are 80%/20% splits of the u.data into ux.base and ux.test, and all the test data sets were disjointed. The results indicate that there is a limit in each data set with $k \sim 6–9$, as shown in data in S9 File.

To further test the existence of this limit, we used data set u.data to calculate the distance matrix of the movies, which contained all the rating data, including the test data in ua.test. The blue curve in Fig 9 shows the results, which show an interesting phenomenon that more data can decrease the prediction error only when $k \leq 3$. After that, a prediction based on the complete data set does not exhibit better prediction accuracy, and may even be worse.

One possible reason for the existence of the prediction limit is that more rating values for movies with longer distances are included in the prediction when $k$ becomes larger, which brings useful information and more noise at the same time. Thus, when the data noise is large enough, the benefit of more data will never be notable. The results on data set u.data can prove this even more: more data brings a larger prediction error with the same $k$ value when $k \geq 3$.

The results in Fig 9 were obtained under the condition of $k$ contiguous h-neighbors, which means that the prediction will be discarded once the $x$th ($x \leq k$) h-neighbor has no rating from predicted user $i$. We have investigated in detail how the value of $k$ affects the prediction results, as shown in Fig 10. The plots indicate that with increasing $k$, the number of predictable ratings
decreases. This illustrates that, for prediction $Y_i(\beta)$, an increasing number of movies $m \in M$ have no ratings from user $i$ when $k$ is larger. The results on the complete data set u.data (blue curve) contain about 1000 more predictable ratings with the same $k$, which further proves the analysis. On the other hand, this result also implies that the prediction accuracy can be improved by using a sufficient amount of useful information—the ratings for the h-neighbors of $\beta$ from user $i$.

Based on the above analysis, we introduce another parameter $\eta$ for controlling the depth of the data used for the prediction. With $\eta$, the h-neighbors retrieving rule defined in Eq 8 becomes, here, $l = k + \eta$ h-neighbors are taken contiguously from $M$ until $k = k'$ or $\eta = \eta'$, as shown in Eq 9, where $k'$ is the number of movies with ratings from user $i$, and $\eta'$ is the number of movies without ratings from user $i$. If the condition $\eta' = \eta'$ is satisfied first, the prediction is discarded.

After adding $\eta$, Eq 8 can be expressed in the form

$$Y_i(\beta) = \sum_{k} r_i x_{\beta}^{d_{x \beta}} \cdot x \in M = \left\{ m | \max_{m_k} (d_{m \beta}) \right\}$$

![Image of graph showing the number of predictable ratings vs. k value.

Fig 10. The number of predictable ratings vs. k value.

doi:10.1371/journal.pone.0146541.g010
Fig 11 plots the results of the prediction accuracy with parameter \( \eta \). The curves indicate that the RMSE will obviously rise when the ratings of movies with greater distances are considered. The fluctuation illustrates that \( \eta \) brings more random factors to the results.

Another result that can be expected is that \( \eta \) will reduce the number of discarded predictions, as shown in Fig 12. In other words, \( \eta \) has the function of controlling the prediction recall. Clearly, the prediction test on the complete data set u.data has a higher recall than that on ua.base because additional data increase the opportunities for obtaining \( k \) h-neighbors before \( \eta' > \eta \).

We also investigate the direct influence of the neighbor distance on the difference between the predicted value and the real value. Fig 13 shows the results, in which the distance of each point is the mean value of \( k = 7 \) h-neighbors and \( \eta = 1 \). We can note that most points lie in the area with a distance of \( ~1.8 \)–\( 2.4 \) and difference of \( ~0 \)–\( 1.0 \), and the difference shows the growth trend with increasing distance. The curves of the mean and standard deviation of the difference apparently account for this, and also prove the above analysis. Thus, one can well imagine that, with enough closer movies, the prediction difference could be effectively reduced.

However, in a real system, some users give random ratings at times, and the ratings obtained for movies are always disproportionate. The left parts of the mean curve and standard deviation curve indicate that notable prediction errors still exist even when the average distance between movies is small. As mentioned in [53], the prediction error can never be zero. Thus, it is remarkable that the presented model can eliminate the predictions that are considered to be
impracticable. Furthermore, researches about big data also imply that the prediction error would be further reduced by combining the historical data based prediction with other near-real-time data, such as feedback of users [54, 55].

Conclusions

We investigated one of the famous benchmark data sets—MovieLens, using an empirical method. There have been numerous studies on recommendation algorithms. Our purpose was not to construct a new recommendation algorithm, but to attempt to find some potential regularity, give user preference a description, and then discuss what factors affect the prediction results and how to eliminate impracticable predictions.

We first mapped the users and movies into a bipartite hyper-network using the rating data, and then presented a definition of the distance between movies. In this definition, we introduced two factors, the shrinking factor and stretching factor, to overcome the data-scale issues. We studied the bipartite hyper-network and found that movies can form many close cliques when only the first-order h-neighbors are considered, which shows that users have explicit preferences. We also found that the size of these cliques closely follows a power law, which implies that the numbers of ratings received for movies are inhomogeneous.

We statistically analyzed the rating distribution of movies that form two-member cliques, and found that most users actually have similar opinions on such movies. We further investigated the distribution of the distances between many two-movie pairs in the data set. We found that the distance data could be used to describe user preferences and predict their ratings.
Then, based on these analysis results, we introduced a user preference model with two tunable parameters. Test results indicated that the presented model could reflect a user’s preference and obtain prediction results with remarkable accuracy under the condition of compromising on recall. This also implied that the presented model has the ability to determine whether a prediction is impracticable.

Further data analysis illustrated that the distance between movies is crucial to a user’s opinion prediction. It contains information about the user’s preferences. However, random factors in the data make prediction error inevitable. Thus, it becomes very meaningful to distinguish which predictions can be made more accurate.

In this paper, we have only reported a few statistical characteristics of a limited data set, and introduced some preliminary methods. In the future, we hope to analyze more data to examine the universality of our findings and try to find more regularity in user preferences.

**Supporting Information**

S1 File. User distance matrix and rating matrix of data set u. (ZIP)

S2 File. Nearest k-1 neighbors and adjacency matrix of data set u. (ZIP)

S3 File. Nearest k-2 neighbors and adjacency matrix of data set u. (ZIP)
S4 File. Rating distribution of item and user in data set u.
(ZIP)

S5 File. Common rating numbers and item similarity of data set u.
(ZIP)

S6 File. User distance matrix and rating matrix of data set u1.
(ZIP)

S7 File. Rating distribution of item and user in data set u1.
(ZIP)

S8 File. Common rating numbers and item similarity of data set u1.
(ZIP)

S9 File. RMSE and their average value with different k value.
(ZIP)

Author Contributions

Conceived and designed the experiments: YSZ. Performed the experiments: BS. Analyzed the data: YSZ BS. Contributed reagents/materials/analysis tools: YSZ BS. Wrote the paper: YSZ BS. Plotted the results of experiments and analysis: YSZ. Implemented the software used in experiments and analysis: BS.

References

1. Lü L, Medo M, Yeung CH, Zhang Y-C, Zhang Z-K, Zhou T. “Recommender systems,” Physics Reports, 2012, 519(1): 1–49.
2. Bobadilla J, Ortega F, Hernando A, Gutierrez A. “Recommender Systems Survey,” Know.-Based Syst., vol. 46, pp. 109–132, Jul. 2013.
3. Barragáns-Martínez AB, Costa-Montenegro E, Burguillo JC, Rey-López M, Mikic-Fonte FA, Peleteiro A. “A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition,” Information Sciences, vol. 180, no. 22, pp. 4290–4311, Nov. 2010.
4. de Campos LM, Fernández-Luna JM, Huete JF, Rueda-Morales MA. “Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks,” International Journal of Approximate Reasoning, vol. 51, no. 7, pp. 785–799, Sep. 2010.
5. Al-Shamri MYH, Bharadwaj KK. “Fuzzy-genetic approach to recommender systems based on a novel hybrid user model,” Expert Systems with Applications, vol. 35, no. 3, pp. 1386–1399, Oct. 2008.
6. Bobadilla J, Hernando A, Ortega F, Bernal J. “A Framework for Collaborative Filtering Recommender Systems,” Expert Syst. Appl., vol. 38, no. 12, pp. 14609–14623, Nov. 2011.
7. Zeng W, Shang MS, Zhang QM, Lü L, Zhou T. “Can dissimilar users contribute to accuracy and diversity of personalized recommendation?,” Int. J. Mod. Phys. C, vol. 21, no. 10, pp. 1217–1227, Oct. 2010.
8. Shinde SK, Kulkarni U. “Hybrid personalized recommender system using centering-bunching based clustering algorithm,” Expert Systems with Applications, vol. 39, no. 1, pp. 1381–1387, Jan. 2012.
9. Goldberg D, Nichols D, Oki BM, Terry D. “Using Collaborative Filtering to Weave an Information Tapestry,” Commun. ACM, vol. 35, no. 12, pp. 61–70, Dec. 1992.
10. Candillier L, Meyer F, Boulé M. “Comparing State-of-the-Art Collaborative Filtering Systems,” in Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition, Berlin, Heidelberg, 2007, pp. 548–562.
11. Herlocker JL, Konstan JA, Terveen LG, Riedl JT. “Evaluating Collaborative Filtering Recommender Systems,” ACM Trans. Inf. Syst., vol. 22, no. 1, pp. 5–53, Jan. 2004.
12. Su X, Khoshgoftaar TM. “A Survey of Collaborative Filtering Techniques,” Adv. in Artif. Intell., vol. 2009, pp. 4:2–4:2, Jan. 2009.
13. Wang J, de Vries AP, Reinders MJT. “Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion,” in Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New York, NY, USA, 2006, pp. 501–508.

14. Blattner M, Zhang Y-C, Maslov S. “Exploring an opinion network for taste prediction: An empirical study,” Physica A: Statistical Mechanics and its Applications, vol. 373, pp. 753–758, Jan. 2007.

15. Han J, Kamber M, Pei J. Data Mining: Concepts and Techniques, Third Edition, 3 edition. Burlington, MA: Morgan Kaufmann, 2011.

16. Zhou T, Jiang L-L, Su R-Q, Zhang YC. “Effect of initial configuration on network-based recommendation,” EPL, vol. 81, no. 5, p. 58004, Mar. 2008.

17. Melville P, Sindhwani V. “Recommender Systems,” in Encyclopedia of Machine Learning, Sammut C. and Webb G. I., Eds. Springer US, 2011, pp. 829–838.

18. Ekstrand MD, Riedl JT, Konstan JA. “Collaborative Filtering Recommender Systems,” Found. Trends Hum.-Comput. Interact., vol. 4, no. 2, pp. 81–173, Feb. 2011.

19. Takács G, Pilászy I, Németh B, Tikk D. “Major Components of the Gravity Recommendation System,” SIGKDD Explor. Newsl., vol. 9, no. 2, pp. 80–83, Dec. 2007.

20. Koren Y, Bell R, Volinsky C. “Matrix Factorization Techniques for Recommender Systems,” Computer, vol. 42, no. 8, pp. 30–37, Aug. 2009.

21. Yedidia JS, Freeman WT, Weiss Y. “Constructing free-energy approximations and generalized belief propagation algorithms,” IEEE Transactions on Information Theory, vol. 51, no. 7, pp. 2282–2312, Jul. 2005.

22. Leung CW, Chan SC, Chung F. “A Collaborative Filtering Framework Based on Fuzzy Association Rules and Multiple-level Similarity,” Knowl. Inf. Syst., vol. 10, no. 3, pp. 357–386, Oct. 2006.

23. Lee SK, Cho YH, Kim SH. “Collaborative Filtering with Ordinal Scale-based Implicit Ratings for Mobile Music Recommendations,” Inf. Sci., vol. 180, no. 11, pp. 2142–2155, Jun. 2010.

24. Zhou T, Ren J, Medo M, Zhang Y-C. “Bipartite network projection and personal recommendation.” Phys. Rev. E vol. 76, no. 4 pp. 046115, Oct. 2007.

25. Zhou T, Su R-Q, Liu R-R, Jiang L-L, Wang BH, Zhang YC. “Accurate and diverse recommendations via eliminating redundant correlations,” New J. Phys., vol. 11, no. 12, p. 123008, Dec. 2009.

26. Lazer D, Pentland A, Adamic L, Aral S, Barabási A-L, Brewer D, et al. “Computational Social Science”, Science, vol. 323, no. 5915, pp. 721–723, Feb. 2009. doi: 10.1126/science.1167742 PMID: 19197046

27. Preis T, Moat HS, Stanley HE. “Quantifying Trading Behavior in Financial Markets Using Google Trends”, Scientific Reports 3: 1684, Apr. 2013. doi: 10.1038/srep01684 PMID: 23619126

28. Moat HS, Curren C, Avakian A, Kenett DY, Stanley HE, Preis T, “Quantifying Wikipedia Usage Patterns Before Stock Market Moves”, Scientific Reports 3: 1801, May. 2013.

29. Goel S, Hofman JM, Lahaie S, Pennock DM, Watts DJ. “Predicting consumer behavior with Web search”, PNAS, vol. 107, no. 41, Aug. 2010.

30. Castellano C, Fortunato S, Loreto V. “Statistical physics of social dynamics,” Rev. Mod. Phys., vol. 81, no. 2, pp. 591–646, May 2009.

31. Galam S. “Minority opinion spreading in random geometry,” The European Physical Journal B, vol. 25, no. 4, pp. 403–406, Feb. 2002.

32. Bordogna CM, Albano EV. “Statistical methods applied to the study of opinion formation models: a brief overview and results of a numerical study of a model based on the social impact theory,” J. Phys.: Condens. Matter, vol. 19, no. 6, p. 065144, Feb. 2007.

33. Lorenz J. “Continuous opinion dynamics under bounded confidence: a survey,” Int. J. Mod. Phys. C, vol. 18, no. 12, pp. 1819–1838, Dec. 2007.

34. http://grouplens.org/datasets/movielens/

35. Berge C. Hypergraphs, Volume 45: Combinatorics of Finite Sets, 1 edition. Amsterdam; New York: North Holland, 1989.

36. Johnson J. Hypernetworks in the Science of Complex Systems. IMPERIAL COLLEGE PRESS, 2014.

37. Gallo G, Longo G, Pallottino S, Nguyen S. “Directed hypergraphs and applications,” Discrete Applied Mathematics, vol. 42, no. 2–3, pp. 177–201, Apr. 1993.

38. Ahn Y-Y, Ahnert S-E, Bagrow JP, Barabási A-L. “Flavor network and the principles of food pairing,” Sci. Rep., vol. 1, Dec. 2011.

39. Newman MEJ. “The structure of scientific collaboration networks,” PNAS, vol. 98, no. 2, pp. 404–409, Jan. 2001. PMID: 11149952
40. Holme P, Liljeros F, Edling CR, Kim BJ. “Network bipartivity,” Phys. Rev. E, vol. 68, no. 5, p. 056107, Nov. 2003.
41. Agarwal S, Branson K, Belongie S. “Higher Order Learning with Graphs,” in Proceedings of the 23rd International Conference on Machine Learning, New York, NY, USA, 2006, pp. 17–24.
42. Collins KD. “Cayley-Menger Determinant.” From MathWorld—A Wolfram Web Resource, created by Eric W. Weisstein. Available: http://mathworld.wolfram.com/Cayley-MengerDeterminant.html.
43. Gritzmann P, Klee V. §3.6.1 in “On the Complexity of Some Basic Problems in Computational Convexity II. Volume and Mixed Volumes.” In Polytopes: Abstract, Convex and Computational (Ed. Bisztriczky T., McMullen P., Schneider R., R.; and Weiss A. W.). Dordrecht, Netherlands: Kluwer, 1994.
44. http://www.cytoscape.org/index.html
45. Zlatić V, Ghoshal G, Caldarelli G. “Hypergraph topological quantities for tagged social networks”, Phys. Rev. E, vol. 80, no. 3, pp. 036118, Sep. 2009.
46. Shang M-S, Lü L, Zhang Y-C, Zhou T. “Empirical analysis of web-based user-–object bipartite networks”, EPL, vol. 90, no. 4, pp. 48006, Jun. 2010.
47. Lambiotte R, Ausloos M. “Uncovering collective listening habits and music genres in bipartite networks”, Phys. Rev. E, vol. 72, no. 6, pp. 066107, Dec. 2005.
48. Grujić J. “Movies recommendation networks as bipartite graphs”, Lecture Notes in Computer Science, vol. 5102, pp. 576–583, Jun. 2008.
49. Vozalis MG, Margaritis KG. “Using SVD and demographic data for the enhancement of generalized Collaborative Filtering,” Information Sciences, vol. 177, no. 15, pp. 3017–3037, Aug. 2007.
50. Lemire D, Maclachlan A. “Slope One Predictors for Online Rating-Based Collaborative Filtering.” In SDM, vol. 5, pp. 1–5. 2005.
51. Gan M, Jiang R. “Constructing a user similarity network to remove adverse influence of popular objects for personalized recommendation,” Expert Systems with Applications, vol. 40, no. 10, pp. 4044–4053, Aug. 2013.
52. Choi K, Suh Y. “A new similarity function for selecting neighbors for each target item in collaborative filtering,” Knowledge-Based Systems, vol. 37, pp. 146–153, Jan. 2013.
53. Hill W, Stead L, Rosenstein M, Furnas G. “ Recommending and Evaluating Choices in a Virtual Community of Use,” in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 1995, pp. 194–201.
54. Preis T, Moat HS. “Adaptive nowcasting of influenza outbreaks using Google searches”, R. Soc. open sci., 1: 140095, Oct. 2014. doi:10.1098/rsos.140095 PMID: 26064532
55. Lazer D, Kennedy R, King G, Vespignani A. "The Parable of Google Flu: Traps in Big Data Analysis", Science, vol. 343, no. 6176, pp. 1203–1205, Mar. 2014. doi: 10.1126/science.1248506 PMID: 24626916