A Novel Recommendation Strategy for User-based Collaborative Filtering in Intelligent Marketing

Jing Yi 1, Liang Zhang 1, Phelan, C. C. 2
1 School of Economics and Management
Guizhou Normal University
Guiyang, 55001, China
2 School of Computer Science
ABBD Co., Ltd. Ottawa, Canada
ftygygz@vip.sina.com

ABSTRACT: Collaborative filtering (CF) is the most successful and widely utilized recommendation technology. CF-based recommenders perform well in terms of accuracy, but they lack the capability to find fresh and novel items. To improve the novel recommendation of user-based CF, the definition of a novel item was established, and an appropriate strategy of novel recommendation was determined. First, a novel item containing the three aspects of likability, unknown, and dissimilarity was defined based on the meaning of the term novel. Second, metrics that measure the novelty of the recommendation system were designed based on the timeliness of novelty. Finally, for the comparison of different strategies of novel recommendation, unknown and dissimilarity were integrated into the process and output of traditional algorithms. Results show that the novelty of the recommendation system is significantly improved when unknown and dissimilarity are integrated into the recommendation results of the traditional algorithm to recalculate the novelty of the item and set the accuracy threshold. Output integration strategy can improve the novelty of the recommended results and can be utilized for any algorithm.

1. Introduction

Recommender systems (RSs) are popular facilities widely deployed to address the challenge of overwhelming information. The recommendation task can be viewed as a prediction problem, in which the system attempts to predict the relevance of certain items to a user and then sorts the items according to the relevance values provided. The importance of an item is generally represented by a numerical value that reflects the degree of user interest during an interaction. The result of a recommendation system is usually a set of items ordered in a descending manner according to the importance scheduled for a given user [1].

Many recommendation algorithms, such as collaborative filtering (CF) [2], content-based recommendation [3], mass diffusion [4], and tag-based filtering [5], are currently applied in recommender systems. CF is the most successful and widely utilized recommendation technology in e-commerce recommender systems. This technology has been widely applied in many e-commerce web sites, such as Amazon, and other applications, such as mobile commerce [6], e-learning [7], and digital libraries [8].

In most cases, a recommendation list contains numerous similar items in the CF algorithm. A major issue of accuracy metrics is their incapability to capture the broad aspects of user satisfaction and the concealment of several blatant flaws in existing systems. Torres suggested that dissimilarities in language and cultural background influence user satisfaction [16]. Swearingen and Sinha examined how usefulness, novelty, and usability are related to user satisfaction through a questionnaire survey.
and found that they are significantly correlated [17]. User satisfaction with recommender systems is related not only to how accurately the system recommends, but also to how much the system supports the decision making of users. Users are satisfied when the recommender system suggests unexpected items that are relevant to their preferences. Recommending relevant items alone often cannot satisfy user expectations. Thus, other characteristics, such as diversity, novelty, and serendipity, must also be considered.

2. Related Works

2.1 User-based CF in recommendation systems

Computation of the similarity between user profiles (user rating vectors) is one of the core components of the user-based (k-nearest neighborhood-based) CF algorithm. Bellogin et al. restated the trust-aware recommendation problem and generalized this problem in terms of performance prediction techniques; they also investigated the adoption of the preceding generalization to define a unified framework, with which they conducted an objective analysis of the effectiveness (predictive power) of neighbor scoring functions and empirically compared four neighbor quality metrics and thirteen performance predictors [9]. One of the main problems of CF is data sparsity. Thi Thuan proposed an enhanced user-based CF algorithm with users’ latent relationship weighting to handle sparse data [10]. Ge Feng proposed a user-based CF recommendation algorithm based on folksonomy smoothing. The approach fills the empty slots through folksonomy technology and then produces recommendations by employing the user-based CF algorithm [11]. In top-N recommendations, sparse users actually perform better than several other categories of users when a standard approach is utilized. The link between sparsity and performance is also considered in the case of predictions. Redpath et al. provided the motivation for targeting improvement approaches toward distinct groups of users as opposed to the entire dataset [12]. To obtain high prediction accuracy, Zhang Zhipeng et al. presented a new model by using covering-based rough set theory to improve CF. In this model, the relevant items of each neighbor are regarded as comprising a common covering. All common coverings comprise a covering for an active user in a domain, and covering reduction is implemented to remove redundant common coverings [13]. Son Le Hoang presented a systematic mathematical definition of fuzzy recommendation system (FRS), including theoretical analyses of algebraic operations and properties, and proposed a novel hybrid user-based fuzzy CF method that integrates the fuzzy similarity degrees between users based on demographic data. The hard user-based degrees are calculated from the rating histories into final similarity degrees to obtain high prediction accuracy [14]. Despite the fact that CF systems are widely used, traditional CF techniques cannot track the preferences of users over a period of time. For this reason, the concept of “temporal dynamics” has become important in recommendation systems. Bakir et al. proposed a new method of providing customized suggestions to users whose tastes may have changed over time. The proposed system is different from traditional user-based and item-based CF techniques because it examines the dates users ranked products and uses these dates to help determine user preferences [15].

2.2 Novel Recommendation

Novel recommendation has recently attracted much attention from both the academy and industry. Novelty has no unified definition. Novel recommendation is simply understood as recommending items that are unknown to users; hence, the simplest means to provide a novel recommendation is to filter items in the profile of the user. Although this method is simple and requires minimal system resources, the items in the user profile account for only a small proportion of all the items. Thus, the result is less effective. Ensuring a certain degree of accuracy for increased novel recommendation is the focus of scholars. The algorithm based on popularity is the most currently used method. G. Shani suggested that novelty can be considered by using an accuracy metric, in which the system does not obtain the same credit for correctly predicting popular items as it does when it correctly predicts non-popular items [18]. Such an adjustment can increase the possibility of recommending unpopular items. Ziegler et al. [19] also utilized accuracy measures that consider popularity. Jinoh Oh proposed an efficient novel recommendation method called personal popularity tendency matching (PPTM), which recommends novel items by considering the PPT of an individual [20]. Herlocker suggested the creation of a list of “obvious” recommendations and removing the obvious ones from each recommendation list before presenting the list to users. A disadvantage of this approach is that the list of obvious items might be different for each user because each person has had different experiences in the past. An alternative approach would be to combine what is known about the user’s tastes with what is known about the community’s tastes [21]. Y. Hijikata suggested the use of the ratings of acquaintances to calculate the probability that a user knows an unrated item [22]. Weng et al. proposed a taxonomy-based recommender system that utilizes hot topic detection using association rules to improve the novelty and quality of recommendations. To better capture the range of user tastes [23], Mi Zhang and Neil Hurley proposed to partition the user profile into clusters of similar items and compose a recommendation list of items that matches each cluster well rather than the entire user profile; they also suggested the evaluation of a number of partitioning strategies in combination with a dimension reduction strategy [24]. Traditional recommendation algorithms only consider the contributions of similar users. Thus, they tend to recommend popular items for users. W. Zeng proposed an recommendation algorithm by considering both the effects of similar and dissimilar users under the framework of CF. The algorithm removes popular items to some extent [25]. Kiboem Lee and Kyogu Lee proposed a new graph-based recommender system that utilizes only positively
In this work, we investigated the definition and strategies
of novel recommendation in user-based CF. The remainder
of this paper is organized as follows. Section 2 reviews
related work on novelty in recommendation systems.
Section 3 presents the definition of item novelty, the design
of an offline experiment and novelty metrics of the
traditional user-based CF, and a strategy of novel
recommendation in user-based CF. Section 4 presents
the results, analyses, and discussion on how the threshold
of accuracy and the weight of the three indicators influence
novel recommendation. Section 5 summarizes the
conclusions.

3. Methodology

3.1 Definition of Novelty and Design of an Offline
Experiment

3.1.1 Definition of Novelty

The definition of novelty plays a crucial role in and is a
basis of novel recommendation. From the research
analysis presented above, researchers define novelty
differently depending on research field and data
characteristics. The algorithm design of novel
recommendation is also different.

According to Wordnet dictionary (http://
wordnet.princeton.edu), the term novel (adj.) has two
definitions: “new – original and of a kind not seen before”
and “refreshing – pleasantly new or different.” Similarly,
the term familiar (adj.) is defined as “well known or easily
recognized.” According to the definition of novel, “novel
items” should have the following three characteristics.

(1) Unknown: The item is unknown to the user

(2) Satisfactory: The item is satisfactory to the user

(3) Dissimilarity: The item is dissimilar to items in the
profile of the user.

A recommendation system explicitly shows that
information regarding the unknown and satisfactory
aspects of items will seriously destroy user experience.
Thus, we only infer the possibility of unknown and
satisfactory aspects of items through the user profile.
We suppose that $dis(i, I_u)$ is dissimilarity between item $i$ and
the set of items in the user profile. Accordingly, the
definition of novelty is as follows:

$$Novelty(i,u) = p(|\text{like},u) \times p(|\text{unknown},u) \times dis(i,I_u)$$  \hspace{1cm} (1)$$

The traditional recommendation algorithm mainly forecasts
the possibility of the user liking an item by modeling the
user’s preferences. This algorithm considers the first
aspect of novelty. In this work, we gradually embed
dissimilarity and unknown variables in the traditional
algorithm to explore methods of enhancing the novel
recommendation of user-based CF.

At present, many researchers utilize popularity to measure
the unknown metric of an item; the lower the popularity
of an item is, the smaller the probability that the item is
known [28–30]. In this work, popularity ($pop_i$) is defined
as the number of items rated, and $p(|\text{unknown},u)$ is
calculated with Formula 2.

$$p(|\text{unknown},u) = \frac{1}{\log(2 + pop_i)}$$  \hspace{1cm} (2)$$

Many methods can be utilized to calculate dissimilarity.
For example, Y. Zhang provided three means to measure
dissimilarity: setting dissimilarity, geometric distance,
and distributional similarity [31]. Two methods can be adopted
to measure the novelty of an item to the user: average
distance (Formula 3) or minimum distance (Formula 4)
between the item and the other items in the user’s profile.
In this work, we adopted minimum distance.

$$dis(i, u) = \min_{j \in I_u}(|i - \cos(i, j)|)$$ \hspace{1cm} (3)$$

$$dis(i, u) = \min_{j \in I_u}(|i - \cos(i, j)|)$$ \hspace{1cm} (4)$$

3.1.2 Offline experiment on novel recommendation

In a traditional experiment, researchers conceal a part
of rated data and use recall and precision metrics to evaluate
the result of the recommendation system. Hidden items
are assumed to meet the user’s preferences. The following
points are ignored: (1) the item that meets the user’s
preferences is not always needed by the user and (2)
hidden items are known to users. The ideal state is that
the recommended item is the user’s next need. Therefore,
the experiment design and evaluation metrics must
consider the factor of time. Therefore, we used a user
behavior dataset with a timestamp and set a time point to
be divided into two subsets. The items rated before the
time point are known to users; thus, recommending these
items is meaningless. The items rated with a high score
by the user after the time point can be assumed as novel
to the user. According to these ideas, we designed a
detailed experimental scheme for novel recommendation
(Figure. 1) and evaluation metrics (Formula 5).

$$Novelty = \frac{1}{|U|} \sum_{u \in U} \left(2 \cdot \text{Recall}_{I_u} - \text{Recall}_{I_u} \right)$$

$$= \frac{1}{|U|} \sum_{u \in U} \left( \frac{2 \cdot \text{Recomm}_{u} \cap \text{Hide}_{bu} \cap \text{Recomm}_{u} \cap \text{Hide}_{bu}}{|\text{Hide}_{bu}|} \right)$$ \hspace{1cm} (5)$$

where $\text{Recomm}_{u}$ is the recommended item set of user $u$,
$\text{Hide}_{bu}$ and $\text{Hide}_{bu}$ are hidden item sets of user $u$.$\text{Recall}_{bu}$
is the recall of traditional experiment, and $\text{Recall}_{bu}$
represents the accuracy metric of predicting the user’s de-
mand in the future. The number of items in the recom
recommendation list is limited. Accordingly, we designed the recommendation list to include more items that meet the future need of the user and to include fewer items known to the user (implication of the metric "novelty"). Meanwhile, the average popularity (Formula 6) and coverage (Formula 7) serve as reference metrics of novelty.

\[
\text{Avg}_{\text{pop}} = \frac{\sum_{i \in U} \sum_{u \in \text{Recomm}} \text{pop}_i}{\sum_{i \in U} |\text{Recomm}|}
\]

\[
\text{Coverage} = \frac{\left| \bigcup_{u \in U} \text{Recomm}_u \right|}{|I|}
\]

where \(\text{pop}_i\) is the rated times of item \(I\), \(U\) is user set, and \(I\) is item set. The two metrics measure the capability of the recommended items and reflect novel recommendation.

In our experiment, we used the MovieLens RecSys2011 data set. This set includes 855,598 rated data of 2,113 users for 10,197 films with a timestamp. A total of 405 rated data of each user and 84.6 rated data of each movie on average are available, and the rated score range is from 1 to 5 points. Movies with a high score are well-liked by users. We utilized the last 300 days of data as \(R_a\) and the rest of the data as \(R_b\). \(R_b\) included 641,600 rated data, and \(R_a\) (which excludes users and movies hidden in \(R_b\)) included 36,266 rated data. We randomly hid 36,266 rated data in \(R_a\) and used them as training data. The hidden data in \(R_a\) and \(R_b\) were used as test data in our experiment. To eliminate the influence of the dataset on the experimental results, we also used the Last FM database in our experiments. The data partition and hiding methods are similar to those mentioned above. The test results are presented as mean values after repeating the experiment five times.

3.2 Traditional user-based CF
In the traditional user-based CF algorithm, \(\{R(u, i, t)\}\) is converted to a UI matrix. Afterward, the similarity between users is calculated.

Lastly, the items liked by the nearest neighbor of the user are recommended. We used Pearson’s correlation coefficient (Formula 8) to calculate the similarity between users Formula 9 to predict the score. Finally, \(N\) items are recommended to the user according to the rank of the predicted scores.

\[
\text{sim}(x, y) = \frac{\sum_{i \in \text{I}(x, y)} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i \in \text{I}(x, y)} (x_i - \bar{x})^2} \sqrt{\sum_{i \in \text{I}(x, y)} (y_i - \bar{y})^2}}
\]

\[
\text{predict}(i, u) = \text{arg}_{\text{r}_u} \sum_{k \in \text{neigh}(u, K)} \frac{\text{sim}(u, k) \text{r}_u}{\sum_{k \in \text{neigh}(u, K)} \text{sim}(u, k)}
\]

\[
I(x, y) = |I_x - I_y|
\]

\[
\text{Recomm}(N, u) = \{i \in I | \text{rank}(i, \text{predict}(i, u)) \leq N\}
\]

where \(\text{arg}_{\text{r}_u}\) is the average rated score of user \(u\), \(\text{neigh}(u, K)\) represents \(K\) users that are most similar to user \(u\), \(r_{ik}\) is the score of item \(I\) rated by user \(u\), \(I_x\) and \(I_y\) represent the rated items of users \(x\) and \(y\). \(\text{Rank}(i, \text{predict}(I, u))\) is the predicted score ranking of item \(I\), and \(\text{recomm}(N, u)\) is the recommended \(N\) items for user \(u\).

We utilized the result of the traditional algorithm as a reference in the following experiments. The results of the following experiments are presented to use Formula 12 to calculate its amplitude of change. In Formula 12, \(M_a\) and \(M_s\) represent new experimental data and reference test data, respectively.

\[
M = \frac{M_a \cdot M_s}{|M_s|}
\]

3.3 Novel Recommendation Strategy
Traditional algorithms are all accuracy oriented; that is, they generally recommend the items preferred by
composite users and focus on the users' preference for recommended items. Judging from its definition, novelty includes three aspects: likeability, unknown, and dissimilarity. Therefore, unknown and dissimilarity can be integrated into the calculation process of traditional algorithms to determine whether the degree of novelty for the recommended items can be improved. Unknown and dissimilarity can also be added to the output of traditional algorithms to determine the novelty of the item. In the following experiments, the user-based CF algorithm with a neighbor number of 20 was adopted to explore the degree of improvement brought by different novel recommendation strategies to the traditional algorithm.

3.3.2 Output Integration
In the output integration, the calculation process of the traditional recommendation algorithm is retained, but new conditions are added to the recommendation results for optimization. The recommendation results of the user-based CF algorithm were adopted to show the users' likability and integrate unknown and dissimilarity to calculate the novelty of the item as the recommendation basis. According to the definition of novelty, Formula 16 was adopted to calculate the novelty of the item.

$$\text{Novelty}(i,u) = \frac{p(i|\text{like},u) \times \min(1, \cos\theta(i,j))}{\log(2 + \text{pop})}$$

(3) Integrate both popularity and dissimilarity
The above analyses suggest that reducing the weight of items with high popularity helps improve accuracy and that increasing the weight of items with high dissimilarity helps improve novelty. Doing both can improve the average popularity and coverage of the recommendation results. In the latter part, the two variables were integrated into the calculation of user similarity (Formula 15).

$$\text{sim}(x,y) = \text{pearson}_\_{\text{pop}}(x,y)$$

$$\text{sim}(x,y) = \frac{\sum_{i \in I(x,y)} \log(2 + \text{pop}) (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i \in I(x,y)} (x_i - \bar{x})^2 \sum_{i \in I(x,y)} (y_i - \bar{y})^2}}$$

3.3.2 Output Integration
In the output integration, the calculation process of the traditional recommendation algorithm is retained, but new conditions are added to the recommendation results for optimization. The recommendation results of the user-based CF algorithm were adopted to show the users' likability and integrate unknown and dissimilarity to calculate the novelty of the item as the recommendation basis. According to the definition of novelty, Formula 16 was adopted to calculate the novelty of the item.

$$\text{Novelty}(i,u) = \frac{p(i|\text{like},u) \times \min(1, \cos\theta(i,j))}{\log(2 + \text{pop})}$$

4. Result Analysis and Discussion
Several strategies of novel recommendation using user-based CF were compared through an experiment. The MovieLens RecSys2011 dataset and the Last.FM dataset were adopted in the experiment, and the number of neighbors was 20.

4.1 Novel recommendation of the traditional user-based CF algorithm
Tables 1 and 2 present the experimental result of the traditional user-based CF algorithm when the number of neighbors is 20. Although the traditional CF algorithm can suitably reflect the user's preferences, the items known to the user occupy a large part. The capability to predict the user's need is relatively weak, and such gap becomes large with the increase in the number of recommendations. These phenomena cannot be observed if the traditional accuracy metric is used. Avg_pop also reveals that traditional algorithms seriously tend to recommend popular items. The duplication of the recommended items for different users is high and resulted in low coverage. Accuracy exhibited a certain degree of improvement with the increase in the number of neighbors; meanwhile, novelty showed no obvious change, and coverage decreased significantly.

4.2 Process integration strategy
The metrics unknown and dissimilarity were integrated
into the calculation of user similarity when process integration strategy was used in user-based CF. The experimental results of the process integration strategy are shown in Figures 2 and 3. When calculating the user similarity, reducing the weight of items with high popularity significantly improved the coverage and effectively decreased the average popularity of the recommendation results. However, accuracy improved by merely 1%–2%, and the improvement in novelty was even less obvious.

Accuracy decreased slightly when the weight of items with high dissimilarity was increased. However, Recall_b decreased more significantly than Recall_a, leading to a slight increase of novelty. In addition, average popularity and coverage improved significantly and are superior to those of the recommendation strategy, in which the weight of items with high popularity was reduced.

When both popularity and dissimilarity were integrated, the metrics were not significantly improved. Accuracy decreased to less than that of the recommendation strategy, in which only the weight of items with high similarity was increased. However, the novelty improvement decreased slightly, and the degrees of improvement for average popularity and coverage of the recommendation results were unobvious.

Comparison of Figures 2 and 3 indicates that the integration of popularity and dissimilarity into the user-based CF algorithm effectively improved the average popularity and coverage of the recommendation results. Reducing the weight of items with high popularity helped improve accuracy but did not significantly enhance novelty. On the contrary, increasing the weight of items with high dissimilarity improved novelty but had an adverse effect on accuracy. After integrating the two metrics, the experimental results still did not improve significantly. We only used unknown and dissimilarity to improve the traditional algorithm. Thus, the calculation of the recommendation results was still accuracy-oriented, and the novelty of the item was indirectly calculated. Therefore, the novelty of the recommendation results did not change fundamentally, but their average popularity and coverage were improved.

<table>
<thead>
<tr>
<th>N</th>
<th>Recall_a</th>
<th>Recall_b</th>
<th>Novelty</th>
<th>Avg_pop</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.0495</td>
<td>0.0962</td>
<td>0.0027</td>
<td>837.2480</td>
<td>0.0743</td>
</tr>
<tr>
<td>40</td>
<td>0.0781</td>
<td>0.1500</td>
<td>0.0061</td>
<td>747.5154</td>
<td>0.0828</td>
</tr>
<tr>
<td>60</td>
<td>0.1054</td>
<td>0.1895</td>
<td>0.0213</td>
<td>688.3498</td>
<td>0.0912</td>
</tr>
<tr>
<td>80</td>
<td>0.1329</td>
<td>0.2216</td>
<td>0.0443</td>
<td>641.1344</td>
<td>0.0981</td>
</tr>
<tr>
<td>100</td>
<td>0.1549</td>
<td>0.2510</td>
<td>0.0589</td>
<td>601.5736</td>
<td>0.1067</td>
</tr>
</tbody>
</table>

Table 1. Novel recommendation of the MovieLens date set

<table>
<thead>
<tr>
<th>N</th>
<th>Recall_a</th>
<th>Recall_b</th>
<th>Novelty</th>
<th>Avg_pop</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.0458</td>
<td>0.0876</td>
<td>0.0027</td>
<td>115.5046</td>
<td>0.1047</td>
</tr>
<tr>
<td>40</td>
<td>0.0496</td>
<td>0.0904</td>
<td>0.0024</td>
<td>109.2574</td>
<td>0.1105</td>
</tr>
<tr>
<td>60</td>
<td>0.0524</td>
<td>0.0958</td>
<td>0.0025</td>
<td>110.2124</td>
<td>0.1185</td>
</tr>
<tr>
<td>80</td>
<td>0.0557</td>
<td>0.1015</td>
<td>0.0023</td>
<td>105.2578</td>
<td>0.1206</td>
</tr>
<tr>
<td>100</td>
<td>0.0602</td>
<td>0.1045</td>
<td>0.0020</td>
<td>108.6587</td>
<td>0.1275</td>
</tr>
</tbody>
</table>

Table 2. Novel recommendation of the Last.FM date set

The number of recommended list

(a) Recall_a

(b) Recall_b
Figure 2. Comparison of evaluation metric using three strategy of process Integration based MovieLens data set

Table 3. Novel recommendation of the MovieLens data set using definition of novelty

<table>
<thead>
<tr>
<th>N</th>
<th>Recall_a</th>
<th>Recall_b</th>
<th>Novelty</th>
<th>Avg_pop</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>-0.9957</td>
<td>-0.9994</td>
<td>-0.8674</td>
<td>-0.9985</td>
<td>6.8471</td>
</tr>
<tr>
<td>40</td>
<td>-0.9888</td>
<td>-0.9994</td>
<td>-0.7298</td>
<td>-0.9975</td>
<td>6.1236</td>
</tr>
<tr>
<td>60</td>
<td>-0.9883</td>
<td>-0.9991</td>
<td>-0.8921</td>
<td>-0.9965</td>
<td>5.3282</td>
</tr>
<tr>
<td>80</td>
<td>-0.9880</td>
<td>-0.9988</td>
<td>-0.9340</td>
<td>-0.9954</td>
<td>4.9321</td>
</tr>
<tr>
<td>100</td>
<td>-0.9865</td>
<td>-0.9985</td>
<td>-0.9354</td>
<td>-0.9942</td>
<td>4.6995</td>
</tr>
</tbody>
</table>

Table 4. Novel recommendation of the Last.FM data set using definition of novelty

<table>
<thead>
<tr>
<th>N</th>
<th>Recall_a</th>
<th>Recall_b</th>
<th>Novelty</th>
<th>Avg_pop</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>-0.9895</td>
<td>-0.9897</td>
<td>-0.8759</td>
<td>-0.9859</td>
<td>7.1258</td>
</tr>
<tr>
<td>40</td>
<td>-0.9795</td>
<td>-0.9905</td>
<td>-0.7965</td>
<td>-0.9824</td>
<td>7.1157</td>
</tr>
<tr>
<td>60</td>
<td>-0.9758</td>
<td>-0.9954</td>
<td>-0.8895</td>
<td>-0.9862</td>
<td>6.1285</td>
</tr>
<tr>
<td>80</td>
<td>-0.9794</td>
<td>-0.9925</td>
<td>-0.9025</td>
<td>-0.9796</td>
<td>5.2015</td>
</tr>
<tr>
<td>100</td>
<td>-0.9814</td>
<td>-0.9851</td>
<td>-0.9145</td>
<td>-0.9904</td>
<td>4.9584</td>
</tr>
</tbody>
</table>
4.3 Output Integration

Formula 16 was used in the output integration strategy to calculate the novelty of the item as the recommendation basis. As shown in the experimental results in Tables 3 and 4, accuracy decreased by more than 98%. Novelty also decreased sharply, and the improvement in popularity and coverage was insignificant. When Formula 16 was directly used to calculate the novelty of the item, popularity and dissimilarity affected the calculation results. The recommended items were those whose popularity and dissimilarity are close to 1. Hence, accuracy decreased by more than 98%. To make novelty meaningful to the target user, items with low popularity and high dissimilarity should be recommended on the premise that accuracy is maintained at a certain level.

The significance of a novel recommendation to the target user is based on the premise that the accuracy of the
Table 5. Novel recommendation of the MovieLans data set setting the accuracy threshold

<table>
<thead>
<tr>
<th>N</th>
<th>Recall_a</th>
<th>Recall_b</th>
<th>Novelty</th>
<th>Avg_pop</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>-0.2097</td>
<td>-0.4651</td>
<td>8.7594</td>
<td>-0.3858</td>
<td>0.3025</td>
</tr>
<tr>
<td>40</td>
<td>-0.1209</td>
<td>-0.3798</td>
<td>7.4727</td>
<td>-0.3287</td>
<td>0.2714</td>
</tr>
<tr>
<td>60</td>
<td>-0.1024</td>
<td>-0.3289</td>
<td>2.3078</td>
<td>-0.2922</td>
<td>0.2500</td>
</tr>
<tr>
<td>80</td>
<td>-0.0875</td>
<td>-0.2895</td>
<td>1.1031</td>
<td>-0.2623</td>
<td>0.2472</td>
</tr>
<tr>
<td>100</td>
<td>-0.0720</td>
<td>-0.2603</td>
<td>0.7308</td>
<td>-0.2373</td>
<td>0.2137</td>
</tr>
</tbody>
</table>

Table 6. Novel recommendation of the Last.FM data set setting the accuracy threshold

<table>
<thead>
<tr>
<th>N</th>
<th>Recall_a</th>
<th>Recall_b</th>
<th>Novelty</th>
<th>Avg_pop</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>-0.1886</td>
<td>-0.4418</td>
<td>8.5248</td>
<td>-0.3524</td>
<td>0.3562</td>
</tr>
<tr>
<td>40</td>
<td>-0.1594</td>
<td>-0.4025</td>
<td>7.2158</td>
<td>-0.3145</td>
<td>0.2954</td>
</tr>
<tr>
<td>60</td>
<td>-0.1324</td>
<td>-0.3215</td>
<td>3.0051</td>
<td>-0.2951</td>
<td>0.2754</td>
</tr>
<tr>
<td>80</td>
<td>-0.1024</td>
<td>-0.3014</td>
<td>1.5241</td>
<td>-0.2634</td>
<td>0.2701</td>
</tr>
<tr>
<td>100</td>
<td>-0.1053</td>
<td>-0.3101</td>
<td>1.0254</td>
<td>-0.2541</td>
<td>0.2573</td>
</tr>
</tbody>
</table>

Table 7. The maximum novelty of the MovieLens data set

<table>
<thead>
<tr>
<th>N</th>
<th>Recall_a</th>
<th>Recall_b</th>
<th>Novelty</th>
<th>Avg_pop</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>-0.2848</td>
<td>-0.6172</td>
<td>11.3889</td>
<td>-0.3762</td>
<td>0.4025</td>
</tr>
<tr>
<td>40</td>
<td>-0.1569</td>
<td>-0.5359</td>
<td>9.1011</td>
<td>-0.3485</td>
<td>0.3215</td>
</tr>
<tr>
<td>60</td>
<td>-0.0971</td>
<td>-0.4436</td>
<td>2.9846</td>
<td>-0.3652</td>
<td>0.2951</td>
</tr>
<tr>
<td>80</td>
<td>-0.1728</td>
<td>-0.4577</td>
<td>1.7521</td>
<td>-0.3784</td>
<td>0.3004</td>
</tr>
<tr>
<td>100</td>
<td>-0.1598</td>
<td>-0.4301</td>
<td>1.1205</td>
<td>-0.3684</td>
<td>0.2814</td>
</tr>
</tbody>
</table>

Table 8. The maximum novelty of the Last.FM data set

recommendation results is maintained at a certain level. Therefore, a threshold should be set for $p(i|\text{like}, u)$. When $p(i|\text{like}, u)$ was greater than threshold $\gamma$, Formula 16 was adopted for calculation; when $p(i|\text{like}, u)$ was less than threshold $\gamma$, the novelty was 0. Given that top-N recommendation was adopted in our experiments, the sequencing method was used for threshold $\gamma$. $N$ items with the highest recommendation scores were selected as an alternative set, among which $N$ items with the highest novelty were selected for recommendation. Tables 5 and 6 show the experimental results when $\gamma$ is equal to 300. Recall_a decreased to far less than Recall_b, causing a very significant improvement in novelty. Thus, this algorithm can effectively rule out the items known to the users, and the decrease in its predictive capability is acceptable. The average popularity and coverage of the recommendation results also improved significantly. Thus, the strategy of setting the accuracy threshold is a wise choice.

The integration of popularity and dissimilarity metrics into the calculation process of the traditional algorithm does not significantly improve the novelty of the recommendation results. However, the output integration strategy significantly improves the novelty when an accuracy threshold is set.

With the increase in the recommendation number, Recall_a significantly decreased in a narrow manner. This finding suggests that the items required by the users...
among the 100 recommendation results are not evenly distributed and that the items ranked at the bottom are more distributed. Thus, better recommendation results can be obtained by increasing threshold $\gamma$. With novelty as the target variable and threshold $\gamma$ as the independent variable, we identified the maximum solution. As shown in the experimental results in Tables 7 and 8, novelty improved slightly. The reason is that Recall_b decreased more significantly than Recall_a, but not so much that the predictive capability was improved as expected.

5. Conclusions

We conducted time-based offline experiments and proposed metrics of novel recommendation. We also compared several novel recommendation strategies on the basis of the definition of novel recommendation. The integration of popularity and dissimilarity metrics into the calculation process of the traditional algorithm did not significantly improve the novelty of the recommendation results. This strategy was limited by the design of traditional algorithms. The output integration strategy adopts the recommendation results of the traditional algorithm as the user’s likability and sets a threshold to maintain the accuracy of the results at a certain level. Unknown and dissimilarity were also integrated into the recommendation results of the traditional algorithm to recalculate the novelty of the item as the recommendation basis. The output integration strategy improved the novelty of the recommendation results and can be applied to all traditional algorithms. However, merely using popularity to measure unknown and minimum distance (or average distance) to measure dissimilarity is too simple. Thus, accuracy decreased sharply. As a result, measuring unknown and dissimilarity in a more scientific manner can be a new research topic in novel recommendation.

References


Electronic Commerce Research and Applications, 10 (1) 94-104.


