A Novel Classification Framework for Evaluating Individual and Aggregate Diversity in Top-N Recommendations

JENNIFER MOODY, University of Ulster
DAVID H. GLASS, University of Ulster

The primary goal of a recommender system is to generate high quality user-centred recommendations. However, the traditional evaluation methods and metrics were developed before researchers understood all the factors that increase user satisfaction. This study is an introduction to a novel user and item classification framework. It is proposed that this framework should be used during user-centred evaluation of recommender systems and the need for this framework is justified through experiments. User profiles are constructed and matched against other users’ profiles to formulate neighbourhoods and generate Top-N recommendations. The recommendations are evaluated to measure the success of the process. In conjunction with the framework, a new diversity metric is presented and explained.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Information filtering

General Terms: Algorithms, Recommenders, Evaluation, Performance

Additional Key Words and Phrases: recommender systems; recommendation accuracy; recommendation quality; performance evaluation metrics; recommendation diversity; collaborative filtering

ACM Reference Format:

1. INTRODUCTION

Recommender systems (RS) have become common place on many e-commerce sites with the aim of helping users to overcome information overload by filtering and suggesting items that may be of interest to them [Resnick and Varian, 1997]. Providing good recommendations has proved difficult, as areas of the dataset are very sparse. The dataset is a central component of the RS. It can be thought of as a set of items that are available for recommendation to a set of users. The information stored on users and items can take the form of ratings, collected either explicitly or implicitly, while other systems may store content features and user demographics. Data sparsity [Adomavicius and Tuzhilin, 2005; Anand and Mobasher, 2005] occurs as the size of the dataset increases as even active users will only have provided ratings for a small percentage of the item set. As a result of this, relationships between users and items can become difficult to infer, in turn making the process of recommendation generation more challenging. A frequently implemented approach for generating personalised recommendations is Collaborative Filtering (CF) [Goldberg et al, 1992; Su and Khoshgoftaar, 2009], whereby the system learns subjective user criteria to construct individual profiles. This technique can be applied using a user-based (UBCF) [Resnick et al, 1994; Shardanand and Maes, 1995] or an item-based (IBCF) [Karypis, 2001; Sarwar et al, 2001] approach.

When implementing UBCF the user-user similarities are calculated based on past behaviours, allowing for users with similar profiles to be grouped together in neighbourhoods consisting of the k-nearest neighbours (NN). UBCF has traditionally been implemented as a memory-based approach and therefore suffers from poor scalability, however it is known for providing users with novel recommendations [Burke, 2002]. In answer to the scalability issues, IBCF was developed. It is commonly implemented as a model-based approach, with the costly item-item similarity calculations completed offline. Unfortunately, IBCF systems suffer from reduced recommendation list diversity [Rashid et al, 2002].

Author’s addresses: J. Moody and D.H. Glass, School of Computing and Mathematics, University of Ulster at Jordanstown, Newtownabbey, Co. Antrim, BT37 0QB, UK.
The key recommendation generation task is to suggest new and interesting items to the users of the system. In general, systems will either predict ratings [Melville et al, 2002] to indicate the user’s expected level of satisfaction for the item in question or select the Top-N [Linden et al, 2003] items from the NN. These approaches can be combined, with the recommendations being ranked according to the predictions and the Top-N presented to the user [Ziegler et al, 2005]. Alternatively the most-frequently-rated (MFR) [Sarwar et al, 2000a] or most-highly-rated (MHR) [Resnick et al, 1994] items in the neighbourhoods are selected as the Top-N recommendations.

Recommender algorithms can be evaluated offline in several ways, with a popular approach being the Leave-N-Out (also known as All-But-N) protocol [Breese et al, 1998]. In this approach each user profile is randomly sampled and a fixed number of items, which are known as the test set (also known as the hidden set), withheld from the system. Fixing the test set size ensures identical user conditions when classification metrics are applied. The items remaining after the removal of the test sets are the items that the system will be trained on and are known as the training set. Another popular approach is K-Fold Cross Validation [Cremonesi et al, 2008], which comprises splitting the dataset into k disjoint folds. After the initial split each fold is used once as the test set while the other k-1 folds are used to train the algorithm. When a high percentage of the items in the test set are uncovered in the recommendations or the algorithm is able to predict the actual ratings accurately, the system is considered to be successful.

In this paper it is proposed that the problem of data sparsity, as identified in the literature [Melville et al, 2002; Anand and Mobasher, 2005], does not affect users in a uniform manner and in fact the users who suffer most in terms of recommendation accuracy when provided with Top-N recommendations are the users deemed to be in the least sparse area of the dataset. Research focused on improving Top-N recommendations [McLaughlin and Herlocker, 2004; Symeonidis et al, 2006] has not identified this problem, and has instead concentrated on addressing issues with the traditional evaluation metrics. It is clear that those users who perform worst when an algorithm has been evaluated are the users who require attention, however in order to target them effectively they must first be correctly identified. The documented problem is a result of the evaluation protocols being biased towards users with a low number of item ratings [Redpath et al, 2010b]. This paper will demonstrate the problem in evaluating Top-N recommendations using a novel user classification framework.

User and item classifications are defined, whereby the user and item sets are split into four mutually exclusive sections in order to ascertain quadrants of users and items at a high level. The user quadrants are employed during recommendation accuracy evaluation, while the item quadrants are used for measuring the diversity of user rating profiles and Top-N recommendations. Accuracy measures are provided, but in light of issues with evaluating only accuracy (discussed in related research), they are set in context by being supplemented by measures for individual diversity of the recommendation set and aggregate diversity, also known as discovery or coverage, of the item set. A diversity index is defined over the item quadrants based on Shannon information.

Once the users’ diversities for rating profiles, and diversity and classification accuracy of recommendation sets are calculated, they are grouped according to the user quadrants and, by doing this, the relationships between the type of user, the diversity of the users’ rating profiles and the diversity and accuracy of their recommendation sets can be established. Surprisingly, it is found that the least diverse user rating profiles result in the most diverse recommendation sets. Interestingly though, this group of users receive the most accurate recommendations but does not contain users from the least sparse area of the dataset.

In order to illustrate the aggregate diversity of the recommendation sets the recommendations are assigned to their respective item quadrants. This in turn allows for the identification of the types of items that are not being recommended. It is proposed that this technique could be employed to increase the aggregate diversity of recommender algorithms by using targeted approaches that cover the types of items that are normally not recommended. Some preliminary testing demonstrates that by simply changing the similarity measure used for creating the NN the aggregate diversity can be greatly increased.
The remainder of this paper is organised as follows; Section Two provides an overview of the related research. In Section Three the methodology is described followed by details on how to create the proposed evaluation quadrants, and apply the accuracy, diversity and coverage measures in Section Four. The Results and discussion are presented in Section Five. The paper finishes with conclusions and future work in Section Six.

2. RELATED RESEARCH

Generating high quality user-centric recommendation sets is the primary goal of a RS. In order to achieve this, there are however, a number of obstacles to overcome: data sparsity, issues with traditional evaluation protocols and metrics, especially the overreliance on accuracy measures, and the evaluation of recommendations as individual standalone items. Improving the diversity of recommendation sets is a recognised method of increasing user satisfaction [Ziegler et al, 2005], and has been extensively researched. Similarly, improving the aggregate diversity of recommendations has been shown to increase user awareness of niche products, leading to an increase in sales of long-tail items [Anderson, 2009]. To date, however, no work, to the authors’ knowledge, has studied the effect of the diversity of the user’s rating profile on, firstly their neighbourhood diversity, secondly, their individual recommendation set diversity and thirdly the aggregate diversity of the recommendations.

Data sparsity is a common problem for recommender systems due to the number of ratings being relatively small and so making prediction difficult [Adomavicius and Tuzhilin, 2005]. This can result in inaccurate neighbourhoods being created as, even if similarities are high, the calculations are potentially based on very few overlapping ratings. Approaches to alleviate the problem of data sparsity have been extensively researched with many new techniques having been suggested; this includes graph-based approaches [Aggarwal et al, 1999], clustering [Ungar and Foster, 1998], hybrid recommenders [Melville et al, 2002] and several dimensionality reduction techniques [Fodor, 2002; Goldberg et al, 2001; Foltz, 1990; Sarwar et al, 2000b; Koren et al, 2009].

There is an abundance of offline evaluation metrics, which are uncorrelated [Herlocker et al, 2004], not standardised for ease of comparison [Redpath et al, 2008] and not widely available as reusable resources [Konstan and Riedl, 1999]. Comparative studies have been conducted [Herlocker et al, 2004; Huang et al, 2007; McLaughlin and Herlocker, 2004; Breese et al, 1998] to evaluate different collaborative recommender algorithms. The metrics used have differed and therefore the degree of improvement in the end results is not transparent.

The research community recognises that the problems with current evaluation methods and measures must be addressed. There are known issues with the standard evaluation methodologies [McNee et al, 2006; Marlin and Zemel, 2009], in that the algorithms are tested on their ability to recommend items or accurately predict a rating for items the user has previously seen, as opposed to evaluating its ability to recommend new and interesting items to a user. This issue has arisen as it is difficult to evaluate offline how good a set of recommendations is if the user has not previously supplied opinions on them. Furthermore, Redpath et al [2010b] discovered when using a standard evaluation method such as Leave-N-Out and traditional classification accuracy measures such as recall that there is a bias towards users with small rating profiles. These users, identified as users in the most sparse areas of the dataset, are not as adversely affected by data sparsity, when evaluated with regards to classification accuracy, as users in dense areas of the dataset. This issue reinforces the need to move away from relying on accuracy measures, as these metrics do not reveal how likely it is for the recommendation set as a whole to satisfy the user’s needs.

Researchers have recognised that in order to achieve user satisfaction we must look beyond simply evaluating recommendation accuracy [Pu et al, 2012; Herlocker et al, 2000; McNee et al, 2006; Swearingen and Sinha, 2002] and that by evaluating the recommendations as individual items, as opposed to sets of recommendations, the recommender community has overlooked the fact that this approach does not always translate into useful recommendations for users [McNee et al., 2006]. For instance, McLaughlin and Herlocker [2004] argue that because mean absolute error (MAE) evaluates each predicted recommendation independently the results are biased towards
algorithms that predict all items well, as opposed to predicting the top items well. The top items, usually 10 or 20, are the items deemed of most interest to the user. Traditionally, recommended items have been evaluated individually, however this does not reflect how users evaluate recommendations: they are inclined to view the recommendation set as a whole [McNee et al, 2006; Ziegler, 2005]. With this in mind and in order to make offline evaluation more user-centered, it is advised that the recommendation set is evaluated as a whole and not as individual unconnected items. Furthermore, other properties, such as individual and aggregate diversity, of the recommendation set must being taken into account during evaluation.

When a user is new to the system, recommending familiar items can be useful in establishing trust, however a set of similar items could be construed as poor recommendations and turn the user off the system. An important aspect in providing a high quality recommendation set, from the point of view of the user, is the diversity of the items contained therein, even if this means a reduction in recommendation accuracy [Pu et al, 2012]. This is especially true if the user is consuming the recommendations in sequential order, for instance when using an online radio station, since they will not be impressed if a number of songs by the same artist play one after the other (unless they specifically requested this). The related research has focused on achieving a balance between the accuracy and diversity trade-off [Smyth and McClave, 2001; McSherry, 2002; McGinty and Smyth, 2003; Ziegler et al, 2005; Adomavicius and Kwon, 2012].

Lack of diversity in case-based and content-based recommendation techniques is a well-known problem [Smyth and McClave, 2001] with too many suggestions in the individual recommendation sets being not only very similar to the target query but also to each other. On the other hand with collaborative filtering the diversity issue is more of a global problem with the majority of recommendation sets containing similar most popular items, therefore reducing the coverage of the item space and ultimately not generating recommendations for long-tail items [Ziegler and Lausen, 2009]. Vargas and Castells [2014] highlighted this issue by showing that the standard neighbourhood method used in collaborative filtering results in a small number of users with large profiles being regularly selected as neighbours therefore reducing the global distribution of items in the recommendations.

The related research has two main foci, firstly to design diversity enhancement techniques, which aim to improve the diversity of recommendations while minimising the expected reduction in accuracy and secondly to provide a measure to capture the achieved diversity [Smyth and McClave, 2001; McSherry, 2002; McGinty and Smyth, 2003; Ziegler et al, 2005; Vargas and Castells, 2011; Adomavicius and Kwon, 2012]. Although numerous researchers have investigated metrics for measuring the diversity of recommendations, a standard accepted metric has yet to be formulated. The proposed metrics take one of two forms measuring either the individual or the aggregate (or global) diversity of the recommendations.

Aggregate measures include measuring the percentage of items that the recommender system ever recommends to users [Herlocker et al, 2004], also known as coverage, counting the total number of distinct items recommended across all users [Adomavicius and Kwon, 2012], as captured by diversity-in-top-N, calculating the entropy of a recommender algorithm [Bellogin et al, 2010] and averaging the uniqueness of users’ recommendation lists for Top-N recommendations, as calculated by Zhou et al [2010]. Individual diversity on the other hand is usually a measure of the average (dis)similarity between the item-pairs of recommendations provided to each user [McGinty and Smyth, 2003; Smyth and McClave, 2001; Vargas and Castells, 2011; Ziegler et al, 2005].

Although it is possible to use a measure such as diversity-in-top-N [Adomavicius and Kwon, 2012] to determine if the level of personalisation is low or high, this metric reveals nothing about the type of items being recommended, i.e. are the items popular or obscure, are they highly correlated with other items? To answer these questions the item quadrants presented in this paper can be used for individual and aggregate diversity evaluation. Furthermore, the current metrics reveal nothing about the user – i.e. are they new or established – do they exhibit popular tastes? These and other questions can be answered using the proposed user quadrants.

As discussed, the diversity of a recommendation set is an important user-centric aspect in relation to recommendation quality. While researchers have proposed diversity enhancement
techniques to improve the recommendation set diversity and various metrics to measure the individual and aggregate diversity of recommendations, no work, to the authors’ knowledge, has captured the role played by the diversity of the initial user rating profile. This paper will examine the effect of the user profile on the diversity of the neighbourhoods, individual recommendation sets and the aggregate diversity.

Previous research has attempted to single out distinct groups of users. For example, Ghazanfer and Prugel-Bennett [2014] identified gray-sheep users by clustering together users with a low similarity to other users and then providing recommendations for this group with a different algorithm. Gray-sheep users [Burke, 2002] are known to receive poor recommendations from collaborative filtering algorithms due to their unusual tastes resulting in them belonging to neighbourhoods with lots of dissimilar users. The evaluation quadrants could be used to identify groups such as the gray-sheep users and evaluate how they perform in comparison to other groups of users, e.g. “power users” [Herlocker et al, 2004]. Cremonesi et al [2008] partitioned the users and items at a ratio of 50:50 based on the number of items a user had rated or on how popular an item was in the dataset. It was found that users with larger profiles obtained better recall results, than users with smaller profiles. They did not however examine users with respect to their similarity with other users.

3. METHODOLOGY
This section describes the methodology of the study and will cover three steps of the personalisation process, namely, Neighbourhood Formation, Top-N Recommendation Generation and Evaluation. User (item) profiles were generated from the datasets described below and matched against other users’ (items’) profiles to formulate neighbourhoods of similar users (items) and generate recommendations for active users. In the final step the recommendations were evaluated to measure the success of the process.

3.1. Dataset
The offline data analysis was based on two rating-based datasets, MovieLens (ML) [MovieLens Dataset, 2006] and BookCrossing (BX) [Ziegler et al, 2005]. Both datasets were transformed into representative user behavioural profiles, the properties of these datasets are outlined in table 1. The ML data was not pre-filtered with all users already having a minimum of 20 ratings, whereas the BX data required pre-processing and had users with less than 10 ratings and books with less than 20 ratings removed.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Domain</th>
<th>Num Users</th>
<th>Num Items</th>
<th>Num Ratings</th>
<th>Rating scale</th>
<th>Rating Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>Movies</td>
<td>6,040</td>
<td>3,696</td>
<td>1,000,199</td>
<td>1-5</td>
<td>Explicit</td>
</tr>
<tr>
<td>BX</td>
<td>Books</td>
<td>5,230</td>
<td>5,174</td>
<td>248,993</td>
<td>0-10</td>
<td>Explicit &amp; implicit</td>
</tr>
</tbody>
</table>

Table 1: Dataset Properties

3.2. Neighbourhood Formation
Experiments were run using both UBCF and IBCF algorithms on ML and UBCF on BX. The neighbourhoods, for both users and items, were formulated using a standard collaborative filtering approach with cosine non-co-rated (COSINENON) as the similarity measure, which is defined in Equation (1),

$$sim(u, v) = \frac{u \cdot v}{\|u\| \|v\|}$$  \hspace{1cm} (1)

where $u$ and $v$ are rating vectors for two users (or items).
By including non-co-rated items [Redpath et al, 2010a] the denominator is able to account for the magnitude of the user’s or item’s rating vector, thus ensuring that users or items with a high number of ratings do not swamp the neighbourhood selections.

3.3. Top-N

The core recommendation generation tasks are to suggest good items to users and/or make predictions that indicate the expected level of user satisfaction [Herlocker et al, 2004]. This paper concentrates on Recommending Some Good Items i.e. generating the Top-N recommendations.

It is assumed that there are many good items that could be of interest to the user. A user-centric approach to recommendation is to provide a small number of good suggestions to prevent the user from having to search through a large set in order to find a handful of good ones. The recommended items at the top of the list, the Top-N, should be the items of most interest to the user. In this report N is set to 10.

The approach adopted uses the MFR [Sarwar et al, 2000a] recommendation generation method. MFR is a simple incremental count of movie rating occurrences. For a given user, the number of ratings of a movie by the neighbours of the user is determined without taking into account the values of the ratings. The Top-N MFR items that do not occur in the active user’s training set are then recommended. In this study the Top-N (N=10) MFR recommendations are not pre-selected [Bellogin et al, 2011]; therefore the recommendations may or may not contain items previously rated by the target user.

Two evaluation protocols were examined, Leave-N-Out and K-Fold Cross Validation. The commonly used Leave-N-Out evaluation protocol was applied to each dataset by randomly selecting a fixed number of items (10 for ML and 5 for BX) from each user’s profile to withhold (referred to as 10Hidden and 5Hidden, respectively). The remainder of the data was used for training purposes. For Cross Validation (CV10) of the accuracy results using ML and BX, K-Folding with parameter k=10 [Whitten and Frank, 2005] was adopted. This required splitting the rating profiles of users into training sets $R^x_T$ and disjoint test sets $T^x_T$, $x \in \{1; 10\}$, at the ratio of 9:1. For both the fixed (Leave-N-Out) and variable (K-Fold Cross Validation) evaluation protocols the hidden items for each user consists of items previously rated by that user.

4. EVALUATION

This section describes how to create the proposed evaluation quadrants, and the accuracy, diversity and coverage measures. Where applicable, the item and user evaluation quadrants are employed to provide a visual representation of the measures. The reader is referred to these publications for more detailed discussions on evaluation [Jannach, et al, 2010; Shani and Gunawardana, 2011].

4.1. Evaluation Quadrants

While there is agreement that sparsity is a problem in recommender systems, the term itself is often used in slightly different ways. An obvious way to characterise sparsity is in terms of number of user ratings, so that a user who has only a few ratings is considered to be more adversely affected by data sparsity than a user with a large rating profile. The same approach can be applied not only to individual users, but to a dataset as a whole so that if a high proportion of user-item pairs have not been rated, users of that dataset are thought to be more adversely affected by data sparsity. The Sparsity Level measure [Sarwar et al, 2001], which will be considered later, is of this kind.

It also seems clear, however, that a user could have a low number of ratings and yet a high proportion of the items rated might also have been rated by other users. In this case the user might well have a high similarity with other users. By contrast, another user might have a high number of ratings yet only a small proportion of the items rated might have been rated by other users. In this case the user might have a low similarity with other users and in spite of having been prolific at rating items may not receive high quality recommendations. In order to take this into account, the approach in this paper will be to consider both the number of ratings and the average similarity with nearest neighbours as contributing to a user’s classification in relation to data sparsity.
There have been various approaches described in the literature which attempt to improve the accuracy of recommendations despite data sparsity [Melville et al, 2002; Ma et al, 2007; Zhou and Luo, 2010]. In terms of evaluating the success of these approaches a common technique is to consider the performance of an algorithm while the ratio of the number of items used in training to that used in testing is varied [Sarwar et al, 2001; Yildirim and Krishnamoorthy, 2008]. The idea is that sparsity is more of a problem when this ratio is low, i.e. when users have a small number of training ratings, and so approaches for alleviating sparsity can be assessed in this case. In contrast, this paper focuses on distinct groups of users in the dataset instead of the dataset as a whole.

In this paper, the sparsity level of a user is understood in terms of two dimensions: first, the number of ratings of the user and, second, the average similarity of the user to other users within their neighbourhood. The approach adopted here is to define an individual user’s level of data sparsity by classifying the user in terms of both of these dimensions simultaneously. The goal then is to investigate the performance of such groups of users in the standard recommendation task of generating Top-N recommendations and to compare the performance of users who are classed as most sparse to those classified as least sparse.

This paper considers that the type of user that would be expected to suffer most greatly from inaccurate recommendations as a result of data sparsity should be defined as a user with few overlapping ratings and a small number of items rated. To identify these users the dataset was divided into quadrants, the boundaries of which were selected based on the median average similarity value and median number of training ratings, see Figure 1, with the axes being “average nearest-neighbour similarity” and “number of training ratings”. The sparsest users are in the low average NN similarity and low number of ratings quadrant, termed Quadrant 1, and the least sparse users in the high average NN similarity and high number of ratings quadrant, termed Quadrant 4. Quadrants 2 and 3 represent intermediate levels of sparsity.

![Fig. 1: Evaluation Quadrants for Categorising Users and Items](image)

The same approach can be used for classifying the item set. For example an item that has a high number of ratings and a high average similarity within its neighbourhood would be assigned to Item Quadrant 4. These items could be thought of as the “Blockbusters” i.e. movies most users have seen and agreed upon the assigned rating. It is proposed that the majority of recommendations will originate from this quadrant. The movies in Item Quadrant 1 on the other hand have not received that many ratings and the users who do rate them tend to disagree. These movies are the long-tail items. They do not necessarily make “bad” recommendations but because of lack of information it is proposed that they will not appear in the recommendation sets that often. In order to provide users with a diverse set of recommendations it is proposed that items should originate from each Item Quadrant. Item Quadrants enable the grouping of items for use during exploratory evaluation of the diversity of user rating profiles, neighbourhoods, and Top-N recommendation sets.

User and Item Quadrants are appropriate for application to any domain that calculates similarities and collects either explicit or implicit ratings. User Quadrants (UQ) provide a means of drilling down into the evaluation of a recommender algorithm, allowing for easy groupings of users, which in turn means groups of users can be targeted with improvement approaches. Item Quadrants
(IQ) provide an evaluation method for measuring individual and aggregate diversity, taking not only the number of ratings an item has received into account but the relationship between items.

Although the quadrants must be recalibrated if the experimental setup changes, i.e. when using a different evaluation protocol, the benefit of this is that they mirror the current information, in other words, they adapt as the dataset changes.

4.2. Accuracy Measures

There is a wide variety of metrics used in the literature to assess the accuracy of a recommender algorithm. In fact, Konstan and Riedl [1999] reports that this abundance of metrics is what makes it so difficult for researchers to evaluate recommender systems. It is important to select the most appropriate metric for evaluating the corresponding recommendation task [Gunawardana and Shani, 2009]. Recommender Algorithms can be evaluated in terms of predictive, classification or rank accuracy [Herlocker et al, 2004]. In this report, recall and precision are the chosen measures for evaluating the classification accuracy of the Top-N Recommendations.

Recall and precision are classified as Successful Decision Making Capacity [Hernández del Olmo and Gaudio, 2008] metrics since their role is to measure the ability of a recommender algorithm to make successful decisions. The evaluation technique used in this study is referred to as Complete (COM), where any recommendation that corresponds to a hidden item is considered a True Positive (TP), i.e. an item from the hidden set which is found in the recommendation set.

Recall is the ratio of TPs to the number of hidden items H, Equation (2), i.e. this metric measures the proportion of successful recoveries from H.

\[
\text{recall} = \frac{TP}{H}
\]

(2)

Precision is the ratio of TPs to the number of recommended items N, Equation (3). In other words precision measures the number of recommendations that are relevant to the user.

\[
\text{precision} = \frac{TP}{N}
\]

(3)

Precision and recall are sensitive to the number of recommendations (N) and hidden (H) items, i.e. as the size of N increases, recall increases and precision decreases. Likewise if H increases, precision increases and recall decreases. When H and N are equal, as they are for the fixed evaluation protocols (10Hidden and 5Hidden) described in section 3.3, precision and recall are equal. Due to the fact that N remains constant in this study even when using cross validation, recall is the chosen reported accuracy metric.

4.3. Diversity Measure

This section will describe and illustrate the new diversity measure being introduced for calculating and exploring the individual diversity of user rating profiles, neighbourhoods and recommendation sets, and the aggregate diversity of the recommendations recommended across all users.

Diversity is measured using the item quadrants. In this paper diversity is quantified using Shannon information, shown in Equation (4). The diversity for a user \( u \) across the item quadrants is given by:

\[
d_u = -\sum_{i=1}^{4} p_i \log_4 p_i
\]

(4)

where the sum is over the four quadrants, \( p_i \) is the probability for a particular quadrant and is defined as the number of items (ratings or recommendations) in that quadrant as a proportion of all the user’s items and base 4 is used for logarithms to ensure that a maximum diversity of 1 is
obtained for a user who has equal numbers of items in each quadrant. Similarly, the aggregate diversity of the recommendations for all users can be calculated using Equation (4) where \( p_i \) is the probability for a particular quadrant and is defined as the number of recommended items from that quadrant as a proportion of all the items in the dataset.

Figure 2 shows an example individual user rating profile spread across the item quadrants.

<table>
<thead>
<tr>
<th>Number of Ratings</th>
<th>Average NN Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>IQ One: 6</td>
<td></td>
</tr>
<tr>
<td>IQ Three: 4</td>
<td></td>
</tr>
<tr>
<td>IQ Two: 5</td>
<td></td>
</tr>
<tr>
<td>IQ Four: 13</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 2: Example individual user rating profile displayed using item quadrants

The diversity index, calculated using Equation (4), for this user’s rating profile is 0.917, indicating that the items rated by this individual are spread out over the item quadrants. This value can then be compared to the diversity value of 0.143 for the user’s corresponding recommendation set, which is illustrated in Figure 3.

<table>
<thead>
<tr>
<th>Number of Ratings</th>
<th>Average NN Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>IQ One: 0</td>
<td></td>
</tr>
<tr>
<td>IQ Three: 1</td>
<td></td>
</tr>
<tr>
<td>IQ Two: 0</td>
<td></td>
</tr>
<tr>
<td>IQ Four: 19</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 3: Example recommendation set displayed using item quadrants

It is clear to see that while the user has rated a wide selection of item types the recommendations do not reflect this.

4.4. Coverage Measures

There are two coverage measures used in this study; user coverage and catalogue coverage. The user coverage (\( U_{cov} \)), as described in Equation (5), is provided for the entire user set and per user quadrant. This allows us to examine the types of users that are not receiving accurate recommendations.

\[
U_{cov} = \frac{\sum_{u \in U} p_u}{|U|} \quad (5)
\]
where $U$ can represent either all the users in the entire dataset or in a given user quadrant and $P_u$ is 1 if the recommendation set of $u$ contains at least 1 successful retrieval from the hidden set and 0 otherwise.

Catalogue coverage ($C_{cov}$) [Herlocker et al, 2004] is defined as the proportion of distinct items recommended across all users to the number of available items. It can be provided for the entire item set and per item quadrant. $C_{cov}$ is an aggregate diversity measure and will be compared to the proposed aggregate diversity measure calculated using Equation (4).

The item quadrants provide a platform from which to examine the types of items that are not being recommended. It is suggested that items belonging to the long-tail [Anderson, 2009] will appear less frequently in the recommendations.

5. RESULTS AND DISCUSSION

In this section the user and item evaluation quadrants created will be described and results, in terms of accuracy, coverage and diversity, presented.

5.1. User quadrant descriptors

The ranges for the user quadrants were defined using the similarity results from COSINENON, and the number of ratings in the active user’s training set. For example, when using this similarity measure on the ML dataset with the evaluation protocol 10Hidden, the median average similarity value was 0.3475 and median number of training ratings 86. This resulted in any user with an average neighbourhood similarity less than 0.3475 and with less than 86 movies rated in their training set being assigned to User Quadrant 1. In comparison, when using the evaluation protocol 5Hidden on the BX dataset the median average similarity value was 0.1781 and the median number of training ratings 42.

The number of users in the quadrant is shown on the user quadrant diagram in Figure 4. As stated in Section 4.1, in this paper, the sparsity level of a user is defined in terms of the number of ratings of the user and the average similarity within their neighbourhood. For the purposes of comparison, the sparsity level as defined by Sarwar et al. (2001), shown in Equation (6), is also displayed for each user quadrant.

$$1 - \frac{\text{nonzero entries}}{\text{total entries}}$$

(6)

It measures the proportion of zero entries in the user-ratings matrix and so takes into account the number of ratings, but it does not take into account the similarity between users. Using Equation (6) and the properties of the ML dataset, as described in Table 1, the sparsity level of the ML dataset is calculated as $1 - \frac{1,000,199}{6040 \times 3,696}$, which is 0.96 (the sparsity level of the BX dataset is 0.99).

5.2. Accuracy and $U_{cov}$

In Figure 4, the average recall, standard deviation of recall and $U_{cov}$ in each user quadrant is reported for 10Hidden UBCF MFR Top-10 recommendations evaluated using the evaluation technique COM (see Section 4.2). Using recall as calculated by Equation (2), a low value is interpreted worse than a higher one. The number of users followed by Sarwar’s sparsity level of each quadrant is shown in brackets after the quadrant number. It is worth noting that the sparsity level scores correspond with the ordering of the quadrants as noted earlier, i.e. Quadrant 1 is the most sparse, Quadrant 4 least sparse and Quadrants 2 and 3 are intermediate.
Since Quadrant 1 represents the most sparse users within the dataset and Quadrant 4 the least sparse (or most dense) users, we would expect performance to be worst for the former and best for the latter. From Figure 4 it can be seen that the users in User Quadrant 1, have actually received the second most accurate set of recommendations and have the second highest $U_{cov}$ value, with an average recall of 0.21 and 87% of users in this quadrant receiving at least one true positive (TP). Quadrant 3, which contains users with a small rating profile but a high similarity with other users, performs the best in terms of recall and $U_{cov}$, with 94% of users receiving at least one TP and the average recall attained being 0.26. Quadrant 4, containing users who have rated a large number of items and have a high similarity with other users, only receives the third highest average recall score and $U_{cov}$ value. The users in Quadrant 2 who have rated a large number of items but have a low average NN similarity are the worst affected in terms of classification accuracy and user coverage with only 74% of users receiving one or more TP and the average recall attained being 0.13. The maximum accuracy score obtained in the user set was 0.70 and the users who obtained this score were located in User Quadrants 1 and 3. The maximum accuracy score received by users in User Quadrants 2 and 4 was 0.60. The $U_{cov}$ across the entire user set was 84%.

Since User Quadrants 1 and 3 outperform User Quadrants 2 and 4 it appears that a greater number of ratings adversely impacts performance. Furthermore, since Quadrant 3 outperforms Quadrant 1 and Quadrant 4 outperforms Quadrant 2 it also seems that users with greater similarity within their neighbourhoods will perform better in terms of classification accuracy. For example, if we move along the NN similarity axis from User Quadrant 1 to User Quadrant 3 we see a 5% increase from 0.21 to 0.26 and similarly if we move from User Quadrant 2 to User Quadrant 4 the increase in recall is 3% from 0.13 to 0.16. If we move along the number of ratings axis from User Quadrant 1 to User Quadrant 2 the decrease in recall is 8% from 0.21 to 0.13 and from User Quadrant 3 to User Quadrant 4 there is a decrease of 10% from 0.26 to 0.16. User Quadrant 3 and User Quadrant 2 contain roughly the same number of users and the sparsity level as defined in Equation (6) indicates that Quadrant 3 is slightly sparser than Quadrant 2. This shows that a greater degree of sparsity does not necessarily correspond to poorer performance, with Quadrant 3 outperforming Quadrant 2, if sparsity is quantified in terms of the sparsity level, as defined by Sarwar et al [2001]. However, the Sarwar sparsity level only takes into account the proportion of non-zero entries in a dataset and does not incorporate the degree of similarity between users, as defined in this paper. When comparing User Quadrant 3 directly with User Quadrant 2, i.e. comparing high average NN similarity with a small rating profile to low average NN similarity with a large profile it can be seen that the recall experiences a 13% jump from 0.13 in User Quadrant 2 to 0.26 in User Quadrant 3. With respect to these findings, it is proposed that if sparsity is to be related with performance, sparsity must be taken to include degree of similarity and the number of ratings. These results suggest that both factors, similarity between users and number of ratings, have an impact on performance.

<table>
<thead>
<tr>
<th>Number of Ratings</th>
<th>Average NN Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>UQ One: (2425 : 0.99)</td>
<td>Recall: 0.21, Std Dev: 0.14, $U_{cov}$: 0.87</td>
</tr>
<tr>
<td>UQ Three: (594 : 0.97)</td>
<td>Recall: 0.26, Std Dev: 0.14, $U_{cov}$: 0.94</td>
</tr>
<tr>
<td>UQ Two: (595 : 0.96)</td>
<td>Recall: 0.13, Std Dev: 0.11, $U_{cov}$: 0.74</td>
</tr>
<tr>
<td>UQ Four: (2426 : 0.91)</td>
<td>Recall: 0.16, Std Dev: 0.12, $U_{cov}$: 0.81</td>
</tr>
</tbody>
</table>

Fig. 4: Average recall, standard deviation of recall and $U_{cov}$ for evaluation of 10Hidden UBCF MFR Top-N using evaluation technique COM. The number of users and Sarwar sparsity level of each quadrant is given in brackets.
To summarise, it has been found that recall is greatest where the users have rated the smallest number of items, decreasing as the user rating profiles become bigger; in addition to this, recall is greatest for those who are within homogeneous neighbourhoods, again decreasing as the neighbourhoods become more heterogeneous. From initial investigations, one possibility for this is that a small user rating profile results in a small candidate item set; and having a homogeneous neighbourhood, in which users are grouped with other users who have rated the same kinds of items in similar ways, results in this smaller candidate item set being less diverse, i.e. it is more probable that the most popular items, which become the recommendations, have been rated by the active user and will therefore appear in the hidden item set. In effect, there is a bias towards users with a small number of ratings [Redpath et al, 2010b].

This result has important consequences for attempts to address the problem of data sparsity. In the context of Top-N recommendation when using the current evaluation protocols, attempts to improve results for sparse users are unlikely to be very successful since sparse users actually perform quite well, which does not appear to have been previously identified in the literature.

To demonstrate how results would be influenced by using a different recommendation generation technique, Highest Weighted Rating (Ziegler, 2005), which produces a prediction for the target item using both the similarity of the nearest neighbours and the rating the neighbours have assigned the target item, was also tested on MovieLens with UBCF. The recall results for each user quadrant are as follows: UQ1: 0.22, UQ2: 0.13, UQ3: 0.31, and UQ4: 0.14. The order of best to worst recall results over the user quadrants described follows the pattern seen with the MFR approach i.e. users in User Quadrant 3 do best overall followed by users in Quadrant 1, then 4, and then 2.

The results from further experiments are presented in table 2. These include the average recall for CV10 of UBCF with ML and BX, UBCF with ML and BX and a fixed number of hidden items and IBCF with ML for both CV10 and 10Hidden.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Dataset</th>
<th>Protocol</th>
<th>UQ1 Recall</th>
<th>UQ2 Recall</th>
<th>UQ3 Recall</th>
<th>UQ4 Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>UBCF</td>
<td>ML</td>
<td>10Hidden</td>
<td>0.21</td>
<td>0.13</td>
<td>0.26</td>
<td>0.16</td>
</tr>
<tr>
<td>UBCF</td>
<td>ML</td>
<td>CV10</td>
<td>0.23</td>
<td>0.13</td>
<td>0.32</td>
<td>0.14</td>
</tr>
<tr>
<td>UBCF</td>
<td>BX</td>
<td>5Hidden</td>
<td>0.050</td>
<td>0.044</td>
<td>0.081</td>
<td>0.070</td>
</tr>
<tr>
<td>UBCF</td>
<td>BX</td>
<td>CV10</td>
<td>0.051</td>
<td>0.045</td>
<td>0.094</td>
<td>0.076</td>
</tr>
<tr>
<td>IBCF</td>
<td>ML</td>
<td>10Hidden</td>
<td>0.15</td>
<td>0.11</td>
<td>0.29</td>
<td>0.15</td>
</tr>
<tr>
<td>IBCF</td>
<td>ML</td>
<td>CV10</td>
<td>0.17</td>
<td>0.12</td>
<td>0.25</td>
<td>0.14</td>
</tr>
</tbody>
</table>

Table 2: Average recall for experiments with UBCF and IBCF algorithms, Fixed and K-Fold Evaluation Protocols and the MovieLens and BookCrossing Datasets.

It can be seen that for every combination in table 2, regardless of the dataset, evaluation protocol or algorithm the users in User Quadrant 3 (highlighted in bold) outperform all the other User Quadrants in terms of recall and the users in User Quadrant 2 always receive the least accurate recommendations. For the majority of results with ML User Quadrant 1 outperforms User Quadrant 4, except for IBCF with 10Hidden where they are equal.

The results are less pronounced for the BX dataset as there is low user coverage. For the entire BX dataset the average U_{cov} is just 21% compared to 84% for the ML dataset. If only the users who receive 1 or more TP are examined, which makes for a fairer comparison in terms of coverage, the results resemble the ML results more closely. For example, the average recall in each user quadrant with BX 5Hidden UBCF is UQ1: 0.24, UQ2: 0.23, UQ3: 0.28 and UQ4: 0.24. Again it can be seen that users in User Quadrant 3 do the best in terms of recall.

As with UBCF ML 10Hidden, when the U_{cov} for the different experiments across the user quadrants is compared it is found that users in User Quadrant 3 receive the highest U_{cov} in the quadrants. For example the BX 5Hidden UBCF U_{cov} results are: UQ1: 19%, UQ2: 18%, UQ3: 28% and UQ4: 27% and the IBCF ML 10Hidden U_{cov} results are: UQ1: 77%, UQ2: 67%, UQ3: 88% and UQ4: 78%.
5.3. Item quadrant descriptors

The Item Quadrants were developed in the same manner as the User Quadrants using the number of user ratings in the training data and average COSINENON (the same similarity measure used for users) neighbourhood similarity. The items were divided into quadrants, with the most sparse items located in Quadrant 1 and the least sparse in Quadrant 4. Quadrants 2 and 3 represent intermediate levels of sparsity. A sparse item can be thought of as an item that has been rated by a small number of users and has a low average NN similarity within its item neighbourhood. For example, when using the 10Hidden dataset split for the MovieLens dataset this resulted in items with less than 119 user ratings and an average NN similarity value of less than 0.2790 being allocated to Quadrant 1. Similarly, items with greater than or equal to 119 ratings and an average NN similarity greater than or equal to 0.2790 were placed in Quadrant 4.

5.4. Catalogue Coverage, $C_{cov}$

The results presented in Sections 5.4 and 5.5 refer to the UBCF ML 10Hidden experiment. The $C_{cov}$ of the Top-N recommendations for MFR COSINENON is shown in Figure 5; the number of items in each quadrant is shown in brackets after the quadrant number and the number of Distinct Recommended Items (DRI) recovered from the recommendations is presented along with the $C_{cov}$. The aggregate diversity, calculated using Equation (4) for the entire dataset, of these recommendations is 0.46.

<table>
<thead>
<tr>
<th>Quadrant</th>
<th>Number of Ratings</th>
<th>DRI</th>
<th>$C_{cov}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>IQ One: (1535)</td>
<td>DRI: 4</td>
<td>4</td>
<td>0.002</td>
</tr>
<tr>
<td></td>
<td>$C_{cov}$: 0.002</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IQ Three: (316)</td>
<td>DRI: 3</td>
<td>3</td>
<td>0.009</td>
</tr>
<tr>
<td></td>
<td>$C_{cov}$: 0.009</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IQ Two: (316)</td>
<td>DRI: 48</td>
<td>48</td>
<td>0.152</td>
</tr>
<tr>
<td></td>
<td>$C_{cov}$: 0.152</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IQ Four: (1529)</td>
<td>DRI: 937</td>
<td>937</td>
<td>0.613</td>
</tr>
<tr>
<td></td>
<td>$C_{cov}$: 0.613</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 5: Number of DRI and $C_{cov}$ for COM evaluation of 10Hidden UBCF with COSINENON MFR Top-N

Globally, 992 distinct items (diversity-in-top-N [Adomavicius and Kwon, 2012]) appeared in the recommendations across all users, which is roughly 27% (coverage [Herlocker et al, 2004]) of the available items. The majority of recommendations, approximately 94%, originated from Item Quadrant 4. Every user, in fact, received no less than 5 recommendations from Item Quadrant 4, with only 4 distinct items being selected from the 1,535 available in Item Quadrant 1, 48 items from 316 in Item Quadrant 2 and 3 items from 316 in Item Quadrant Three. Only 6 users received recommendations from Item Quadrant 1, 124 from Item Quadrant 2 and 14 from Item Quadrant 3. These results demonstrate the global problem, discussed in related research, experienced by collaborative filtering algorithms whereby the recommendation sets are constructed from the most popular items, i.e. Blockbusters, resulting in a reduction of catalogue coverage and ultimately aggregate diversity.

Figure 6 demonstrates how making a simple change, such as using a different similarity measure, can result in a different spread of items, and in this case a considerable improvement to the aggregate diversity and $C_{cov}$. In this example Pearson’s Correlation is used to measure the similarity between users and formulate the NN, again the Top-10 MFR items are selected as the recommendations.
It is interesting to note that of the 3,696 items available, the aggregate recommendation set generated when using Pearson’s Correlation, discovered 1,293 (~35%) distinct items. While the majority of recommendations are still originating from Item Quadrant 4, 35% of the items available from Item Quadrant Three appear at least once in the recommendations and the proportion of items coming from Item Quadrant 1, as captured by $C_{cov}$, has improved considerably, increasing from 0.002 with COSINENON to 0.153 with Pearson’s Correlation. All users received at least 1 recommendation from Item Quadrant 4, 1,490 users received at least one recommendation from Item Quadrant 1, 222 users from Item Quadrant 2 and 1,266 users from Item Quadrant Three. The aggregate diversity for these recommendations was 0.91.

One explanation for the increase in item coverage and aggregate diversity when using Pearson’s Correlation compared to COSINENON is that the neighbourhoods being formulated are more diverse and therefore the candidate item sets contain a greater variety of items to select from. The $C_{cov}$ for Item Quadrant 2 remained relatively unchanged when using COSINENON or Pearson’s Correlation suggesting that items having a low similarity are less likely to occur in recommendations generated using a collaborative filtering technique, despite the fact they have a large number of ratings. If this was an online business that means most of the 1,535 items in Item Quadrant 1 and 316 items in Item Quadrant 2 (roughly 49% of products) may never be suggested to potential customers. It may be reasonable in this case to develop ways of injecting these items into recommendations or to use a different technique to generate sections of the recommendation sets.

5.5. Diversity results

Diversity results were obtained for user rating profiles, neighbourhoods and recommendation sets. The individual diversity of each user’s rating profile and recommendation set was calculated as described in Section 4.3, while the diversity of each neighbourhood is calculated as the average individual diversity of the user rating profiles within that neighbourhood. Finally, the diversity of each group of users (user profile, neighbourhood, recommendation set) is calculated as the average over all the individual diversities of the users within that group.

Figures 7, 8 and 9 display the diversity results. Figure 7 is the average diversity and standard deviation of the diversity of user rating profiles per user quadrant. Figure 8 presents the average diversity of neighbourhoods whereby the diversity of the neighbourhood for each user belonging to that user quadrant is calculated as described above and then the average across each user quadrant is taken. Finally, Figure 9 presents the average diversity and standard deviation of diversity for the recommendation sets per user quadrant.
The average diversity of user rating profiles across the ML dataset was 0.30 while the maximum diversity was 0.92. It is interesting to note that users in User Quadrant 3 have the least diverse user rating profile, while receiving the most accurate recommendations in terms of classification. In fact, the order of increasing user rating profile diversity, UQ3, UQ1, UQ4, UQ2, mimics the decreasing order of classification accuracy as shown in Figure 4. One possible explanation for this is that as the user rating profile becomes more diverse it is harder to predict previously sampled items. This shouldn't be surprising as the more focused a profile is on particular types of items, the easier it should be to predict what other items the user may have rated. This however does highlight an issue with current accuracy metrics such as precision and recall [Redpath et al., 2010a], which measure an algorithm's ability to find hidden items as opposed to providing a measure of how good all the recommended items are.

![Average NN Similarity Table](image)

Fig. 7: The average diversity and standard deviation of the diversity of user rating profiles per user quadrant

Figure 8 demonstrates that the diversity of the neighbourhoods follows the same pattern seen in Figure 7. Quadrant 3 users have the least diverse neighbourhoods and Quadrant 2 users have the most diverse. An initial analysis leads the authors to believe that users with diverse profiles tend to attract neighbours with similar profiles. This will be investigated further in future work where the characterising of user quadrants and exploration of neighbour types is suggested.
Figure 9 shows that the users in User Quadrant 3 have, on average, the most diverse recommendation sets. This result is somewhat surprising since they have the least diverse rating profiles and the most accurate recommendations, although the high accuracy is a result of the evaluation protocols being biased towards users with a low number of item ratings [Redpath et al, 2010b]. The most diverse user recommendation set had a diversity measure of 0.49 and this user belonged to User Quadrant 1. The most diverse recommendation set in User Quadrants 2, 3 and 4 were 0.22, 0.41 and 0.21, respectively. Quadrant 4, which represents 40% of the user set, has the second most diverse user rating profiles but received the least diverse and the third most accurate recommendations. It is proposed that the lack of diversity is because they have rated so many items, including the most popular items in the item set and therefore their recommendations are all very similar. Once again, the lack of accuracy is a result of the bias in the evaluation protocols towards users with a low number of item ratings.

All of the results from examining the accuracy and diversity of UBCF on ML with 10Hidden MFR Top-N as displayed in Figures 4, 7, 8 and 9 are summarised in Figure 10.

From Figure 10 it is clear that the least diverse user profiles result in the most accurate recommendations and vice versa. Somewhat surprisingly, the diversity of the user's initial rating profile is not a direct indicator of how diverse their final Top-N recommendations are likely to be. The reason for this will be explored further in future work.

6. CONCLUSIONS AND FUTURE WORK

This paper has provided a novel definition of sparse users in terms of low average similarity with nearest neighbours and low number of ratings. Furthermore, it has presented a new evaluation framework for recommender systems and justified the need for this framework using a variety of experiments. It has been demonstrated that in contradiction to common assumptions, not all users suffer as expected from the data sparsity problem. As the data sparsity problem is common to all classification problems, the approaches suggested in this study could be worth considering in broader areas. By defining sparse users in this way it has been shown that sparse users do not perform worse for Top-N recommendation when using traditional evaluation protocols. This counterintuitive result has been explained in terms of a bias towards users with smaller rating vectors [Redpath et al, 2010b]. Users with small rating profiles receive the most accurate recommendations, and the accuracy of the Top-N recommendations decreases as the size of their rating profile increases.
Furthermore, a user with a small rating profile who also belongs to a homogeneous neighbourhood will receive more accurate recommendations than a user with a similar sized rating profile who belongs to a heterogeneous neighbourhood. It is hypothesized that one reason for this outcome could be that users generally find themselves in neighbourhoods with users who have similar sized rating profiles. If the active user has a small rating profile, and a high similarity with other users, this results in a small homogeneous candidate item set i.e., the candidate item set is not diverse due to the fact that most of the neighbours have all rated the same popular (for that neighbourhood) items and therefore the likelihood of finding the hidden items belonging to the active user is increased. These results strengthen the argument that recommender systems cannot be robustly evaluated using traditional metrics and methods alone and that new metrics, similar to the diversity metric proposed in Section 4.3, need to be presented along with accuracy results in order for a complete user-centered evaluation to take place.

As a result of examining individual diversity alongside accuracy in the context of the User Quadrants, it is possible to identify and describe groups of users that perform in varied ways for different factors. For example, users in User Quadrant 3 appear to be quite clear on their preferences, i.e. their user rating profiles are not diverse and they belong to tight neighbourhoods, i.e. people who rate the same items in similar ways. They perform well in terms of accuracy and in terms of individual diversity. Work has started on extending the quadrants by increasing the gradations and developing textual descriptions for each User Quadrant in order to characterise user groups. The intention is increased personalisation, whereby each group of users is targeted with the algorithm most suited to their characteristics. For instance, the algorithm used in this study resulted in users from User Quadrant 1 receiving quite accurate recommendations. If User Quadrant 1 is further decomposed we could identify the new users of the system and they may in fact be happiest with the current algorithm, because receiving good accuracy results could improve their trust in the system. Another group that could potentially be identified in User Quadrants 1 and 2 are the gray-sheep users, i.e. users that exhibit unusual tastes and therefore attain a low similarity with other users, as these users traditionally do not do as well with CF algorithms. Once they are identified they could be targeted with other recommendation approaches. Likewise, it may be that certain groups in User Quadrants 2 and 4 do not need their recommendation diversity improved because they like to watch the same kinds of movies all the time. It is intended that these assumptions will be strengthened with a longitudinal user study. Future work also includes the incorporation of further sensitivity testing, for example varying the value of N and exploring the effect of different similarity measures. Additionally, it is intended that future work will examine the results of different recommender algorithms, the development of more individual and aggregate evaluation measures, the gradation and characterisation of Item Quadrants, and the identification of item types that do not appear, often or at all, in the recommendation sets.

REFERENCES


