Evaluating the Relative Performance of Neighbourhood-Based Recommender Systems

Humberto Jesús Corona Pampín¹, Houssem Jerbi², and Michael P. O’Mahony²

¹ CLARITY Centre for Sensor Web Technologies, University College Dublin, Ireland
humberto.corona@ucdconnect.ie

² Insight Centre for Data Analytics, University College Dublin, Ireland
houssem.jerbi@ucd.ie, michael.omahony@ucd.ie

Abstract. Neighbourhood-based recommender systems are a class of collaborative filtering algorithms, which rely on finding like-minded users to generate recommendations, automating what is usually known as word-of-mouth. These systems attempt to solve the information overload problem by presenting the user with relevant items. However, there is evidence showing that these algorithms may contribute to the filter bubble problem, making it harder for the user to find interesting items which are non-popular. In this paper we propose a novel evaluation of the performance and biases of the two most common neighbourhood-based approaches: user k-nearest neighbour collaborative filtering (UKNN), and item k-nearest neighbour collaborative filtering (IKNN). We propose an evaluation which considers the size of the neighbourhood, finding that optimising for accuracy in UKNN algorithms leads to a poor performance in terms of diversity, a higher bias towards popularity, and less unique recommendations, when compared to the IKNN approach.

Keywords: recommender systems, collaborative filtering, diversity, popularity, evaluation

1 Introduction

As access to different types of media becomes easier and the amount of user-generated content increases, it has become very difficult to deal with the large catalogs of information available on the web. For example, finding a new movie to watch, discovering new bands to listen to, or deciding whom to follow in online social networks is no longer a straightforward task. This problem is commonly referred to as information overload. There is need for a new class of intelligent systems that help users to navigate through all this information, by automatically selecting the most relevant content and filtering that considered to be irrelevant. Recommender systems address this problem by learning the preferences of users. The gathered knowledge is then applied to personalise the user experience in different services; from advertising to clothes recommendations, movie recommendations and music discovery.

Extensive work has been carried out on the evaluation of recommender system algorithms, including surveys of the field [8,16]. Nevertheless, there are
The relationship between error-based metrics (such as $RMSE$) and accuracy-oriented metrics (precision and recall) is studied in detail in [4]. The authors explain how error-based metrics are not appropriated to evaluate algorithms that present the user with a list of top recommendations, which is also highlighted in [6,19]. They also explain that improvements in algorithms as measured by $RMSE$ do not translate in improvements in accuracy-oriented metrics.

An evaluation of neighbourhood models is presented in [13], which focuses on the influence of neighbour selection on the performance of $UKNN$ and $IKNN$ algorithms. The authors explain how the performance of the algorithm is biased by the evaluation methodology. They show that neighbourhood selection has a limited effect on performance, and that the contributions of neighbours are often disguised or misleading. Recent work [2] proposes the use of graph partitioning
techniques for neighbour selections, which outperforms state of the art techniques in terms of ranking precision.

In this paper we also analyse the effects of neighbourhood selection on the performance of UKNN algorithms. We propose a new training and test set configuration, and we evaluate recommendation performance using accuracy-orientated metrics in conjunction with a number of other metrics as described in Section 3.

2.2 Beyond Accuracy

Recently, several works such as [9, 10, 14] have focused on the study of diversity. Diversity measures how different are the items recommended in a list. For example, if a set of recommendations contains only movies from the Harry Potter saga, the recommendations may be accurate, but not diverse.

Said et al. [14] use a furthest neighbour model to increase diversity, generating almost orthogonal recommendations, when compared to the traditional UKNN approach. Hurley [9] introduces diversity within the optimisation function in a matrix factorisation approach. The results show more diverse recommendations, without having to post-filter the recommendation lists.

Diversity is also studied in [20], which analyses the diversity problem of top-N recommendations from a binary retrieval perspective. This work tries to maximise both diversity and similarity with respect to the user profile. A new metric for evaluating the proposed diversity approach is also presented. The results show improvements in both accuracy and diversity.

Work presented in [3] analyses the novelty and popularity of recommendations. The work shows that content-based recommender systems (applied to music) are not biased towards popularity, while a standard collaborative approach is perceived as providing higher quality recommendations even when the recommendations are not as novel.

An evaluation of recommender systems coverage and serendipity is explored in [5], which focuses on the contribution of those metrics to the quality of a system. The paper proposes metrics to evaluate the two properties that include the usefulness of a recommended item. The authors argue that the proposed metrics correlate with the real user experience better than previous metrics.

In this paper, together with an analysis of precision, we also evaluate properties such as diversity and popularity. Furthermore, we perform a study on the uniqueness of the recommendation lists generated by different algorithms.

3 Relative Performance of Neighbourhood Models

Neighbourhood-based models are widely used and have been extensively evaluated. However, it is not completely clear how different factors such as neighbourhood selection or item popularity bias affects the overall performance of these algorithms.
We study the effect of neighbourhood selection for the UKNN algorithm, which relies on finding similar users to generate recommendations. More specifically, we experiment with different values for the number of neighbours, $k$, and compare the results with the IKNN approach, which makes collaborative recommendations based on items similar to those the user has liked in the past.

To explore how neighbourhood size affects performance, we study recommendation precision, diversity and popularity. We compare the results for different values of $k$ for two different datasets. By performing this analysis, we are able to understand what is the optimal value for $k$ and what are the tradeoffs between the different properties evaluated.

### 3.1 Evaluation Metrics

Four metrics are chosen as part of our evaluation approach: precision (PRC), popularity, diversity and uniqueness. These metrics, which are independent from the recommendation algorithms, are averaged across all users in the evaluation. The details of these metrics are described below.

**Diversity (DIV)** [18, 21] is evaluated over the ranked list of recommended items, $r_1, r_2, \ldots, r_n$, presented to the user. It captures how different the recommended items are from each other. In this case, diversity is evaluated based on a pairwise comparison of the items in the list (Equation 1). This metric relies on item co-ratings to calculate the similarity between items. Here, similarity ($Sim$) is computed using the cosine metric.

$$Diversity(r_1, \ldots, r_n) = \frac{2}{n|n-1|} \left(1 - \sum_{i<j}^n Sim(r_i, r_j)\right)$$ (1)

**Popularity (POP)** can have several meanings depending on the context of the recommendations. For example, in the music domain, artist popularity is measured by the number of records sold while Twitter identifies trending (i.e. popular) topics using text analysis approaches. We use the number of available ratings per item to calculate item popularity. The popularity metric $POP_i \in [0, 1]$ for item $i \in I$ is defined in Equation 2, which is the total number of ratings for item $i$, $n_i$, divided by the total number of ratings for all items in the system, $n_I$.

$$POP_i = \frac{n_i}{n_I}$$ (2)

**Uniqueness** measures the ability of an algorithm to generate recommendations which are not generated by other algorithms. For example, if we consider two recommendation sets, $R^a$ and $R^b$, which are generated by different algorithms, the uniqueness of the recommendation set $R^a$ can be calculated as the number of items in the set $R^a$ which are not found in $R^b$ (Equation 3).

$$Uniqueness(R^a) = R^a \setminus R^b$$ (3)
Precision (PRC) measures the fraction of recommended items which are relevant (Equation 4).

\[ PRC = \frac{|\text{recommended} \cap |\text{relevant}|}{|\text{recommended}|} \] (4)

3.2 Algorithms

The standard UKNN and IKNN algorithms are considered in this work as described below. Cosine is used as the similarity function in both algorithms.

- **User-based** \( k \)-nearest neighbour (UKNN) leverages the ratings matrix to find users with similar tastes (i.e. neighbours) to the target user [7]. Recommendations are then based on the neighbours’ preferences for items unseen by the target user. In this way, the approach automates the word-of-mouth model, in which recommendations are usually spread in society [17].

- **Item-based** \( k \)-nearest neighbour (IKNN) relies on an analysis of the ratings matrix to identify items similar to those that are liked by the user. The similarity between items is calculated as a function of the ratings they receive from users [12, 15].

4 Experiments

In this paper, we compare two different recommendation algorithms: the standard IKNN and UKNN algorithms, as described in Section 3.2. The former is evaluated with a fixed number of neighbours (\( k = 300 \))\(^3\), while for the latter, neighbourhood sizes in increments of 10 are considered (\( k \in [10, 200] \)). The goal of the evaluation is to understand the effects of neighbourhood size on the performance of the user-based approach, and perform a relative comparison with respect to the item-based approach.

Both algorithms are evaluated in the context of the rank items in the catalog task as described in [8], generating a list of top-10 recommendations per user. Therefore, error-based metrics are not used in the evaluation of the algorithms, they are not adequate in this scenario [8].

To perform the evaluation, the test set for each user contains 10 liked items and top-10 recommendations are made for each user. Hence, the values of precision and recall will be the same in this setting (both given by the intersection between the items in each user’s test set and the recommended items, divided by 10).

The details of the training and test set construction are as follows:

\(^3\) Fixing neighbourhood size at \( k = 300 \) for IKNN allows us to reduce the number of free parameters in the experiment. Moreover, preliminary analysis of the effect of neighbourhood on the IKNN algorithm did not show trends as pronounced as for the UKNN approach. A more detailed analysis of IKNN is left to future work.
– Both of the datasets considered in this work include users with at least 20 rated items in their profile. Only users with at least 10 liked items are included in our evaluation. A rating threshold of 4 is used to decide whether an item is liked or not; i.e. all items which received a rating of at least 4 (out of 5) are considered liked. This particular rating threshold has been used in previous work with similar datasets [1, 11].
– The test set for each user, $T_{e_u}$, is created using 10 randomly selected liked items. In this way it is guaranteed that each user’s test set will have the same number of items, all of which are liked.
– The training set for each user, $T_{r_u}$, is created using all items which are not included in the test set.

4.1 Dataset

To evaluate the proposed approach, two of the MovieLens [7] datasets are selected: the MovieLens-100k (ml-100k), and the MovieLens-1M (ml-1M), containing 100,000 and 1 million ratings, respectively. As mentioned above, the original datasets are processed so that only users with at least 10 liked items are included. Table 1 shows the statistics for both datasets after processing is performed.

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Datasets</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ml-100k</td>
</tr>
<tr>
<td>Number of users</td>
<td>495</td>
</tr>
<tr>
<td>Number of items</td>
<td>1,486</td>
</tr>
<tr>
<td>Number of ratings</td>
<td>76,982</td>
</tr>
<tr>
<td>Ratings per user (mean)</td>
<td>155</td>
</tr>
<tr>
<td>Ratings per user (std. dev.)</td>
<td>108</td>
</tr>
<tr>
<td>Ratings per item (mean)</td>
<td>51</td>
</tr>
<tr>
<td>Ratings per item (std. dev.)</td>
<td>61</td>
</tr>
<tr>
<td>Density</td>
<td>0.105</td>
</tr>
</tbody>
</table>

Table 1: MovieLens dataset statistics after preprocessing.

4.2 Results

In this section we present the results of the approach proposed in Section 3. Recommendations made using both algorithms are based on training set data only. Each user is presented with 10 recommended items which are evaluated based on that user’s test set items. We evaluate the results of the generated recommendations in terms of precision, popularity, diversity and uniqueness, and compare performance as the neighbourhood size, $k$, in the UKNN approach is varied.
With this experimental methodology, we expect to better understand the relationship between the evaluated properties and their dependency on neighbourhood size. Moreover, we wish to analyse the differences between the recommendation lists generated by the two algorithms, and examine how much they differ from each other by measuring average uniqueness.

Figures 1 and 2 show the results of the four properties evaluated in both datasets. Overall, precision, popularity, diversity and uniqueness show very similar trends for both datasets. To begin, precision results are first discussed followed by the remaining properties, focusing on the relationship between each metric and precision.

The results for **precision** are shown in Figures 1a and 2a. The results show that the overall precision of UKNN increases up to \( k \approx 80 \), and thereafter declines for the ml-100k dataset but remains relatively constant for the ml-1M dataset. It is noteworthy that the precision of UKNN exceeds that of IKNN at much smaller neighbourhood sizes; at \( k = 30 \) and at \( k = 15 \) for the ml-100k and ml-1M datasets, respectively.

The results for **recommendation popularity** and **diversity** are shown in Figures 1b and 2b. The results indicate that the average popularity of recommended items increases with \( k \). Thus, as expected, there is a high bias toward popularity at larger sizes of \( k \), and a corresponding decrease in recommendation diversity. In the limit, if every user was considered a neighbour, the recommendations would resemble the distribution of item ratings in the dataset, and thus only the most popular items would be recommended.

The average recommendation diversity decreases as \( k \) increases for the UKNN algorithm, and approaches that of the IKNN algorithm at \( k = 200 \). However, this reduction in diversity for UKNN is not compensated by a corresponding gain in precision at larger neighbourhood sizes. As previously described, the UKNN algorithm attains its maximum accuracy at \( k \approx 80 \), yet the loss in diversity is a monotonically decreasing function. We can conclude that there is a high inverse correlation between popularity and diversity, and both suffer at larger values of \( k \); i.e. recommendation lists contain mainly popular items which have a high degree of similarity to each other.

Finally, Figures 1c, 1d, 2c and 2d depict the average numbers of recommendations which are **unique** to the UKNN algorithm and those which are **common** to both algorithms, and the corresponding precision results over the unique and common recommended items. We can observe that the number of unique recommendations decreases significantly as \( k \) increases. For example, at \( k = 100 \), on average more than 7 of the 10 recommended items produced by both algorithms are the same in the case of the ml-100k dataset. It is also apparent that recommendation accuracy is largely due to the common items which are recommended by both algorithms. Consider again the ml-100k dataset at \( k = 100 \), where the precision of unique and common items is approximately 0.04 and 0.21, respectively. However, it should be noted that the unique items recommended by the UKNN algorithm are not necessarily irrelevant — although these items are not present in the test set (which is used as the ground truth), they may
Fig. 1: Performance vs. neighbourhood size, ml-100k dataset.
Fig. 2: Performance vs. neighbourhood size, ml-1M dataset.
still represent useful recommendations to users. Moreover, at smaller values of $k$, items recommended by UKNN are less popular, and such items – by definition – are less likely to appear in test sets. This is a well-known limitation of the standard approach to the ‘offline’ precision evaluation of recommender systems as performed in this work, and it can only be addressed by live user trials which is left to future work.

In summary, the experimental results indicate the bias inherent in the UKNN algorithm towards popular recommendations at larger neighbourhood sizes, which is more evident in the ml-100k dataset. However, the loss of diversity and bias towards popular recommendations at larger values of $k$ is not compensated by a gain in precision. From the results, we also observe that the uniqueness of the recommendations also depends on $k$, and that smaller neighbourhood sizes lead to more unique, less popular, and more diverse recommendations.

5 Conclusions and Future Work

In this paper we have performed an evaluation of the user-based UKNN and item-based IKNN collaborative recommendation algorithms. We have presented the analysis of our evaluation results, which focuses on the effect of neighbourhood size on the relative performance of the algorithms. We have described the results according to four main properties: precision, diversity, popularity, and uniqueness of the recommendations.

Experiments performed using two different datasets show that optimising for accuracy in the user-based approach leads to a poor performance in terms of diversity, a higher bias towards popularity, and less unique recommendations. Moreover, choosing smaller numbers of neighbours for UKNN leads to more diverse and less popular recommendations, at the cost of a relatively small decrease in accuracy. We have also analysed the differences between the two approaches in terms of unique recommendation capabilities. The experiment demonstrated that when the number of neighbours in the UKNN approach is large, both algorithms tend to produce very similar recommendations (up to 75% and 60% for the ml-100k and ml-1M datasets respectively).

The experiments in this paper show interesting findings in terms of the correlation between accuracy, popularity and diversity. In future work, a study of novel neighbourhood selection algorithms will be considered. Furthermore, we also plan to expand this work considering other properties, such as user and item coverage. Finally, we plan to perform an online evaluation that investigates the correlation between the different properties and real user behaviour.

6 Acknowledgements

This work is supported by Science Foundation Ireland through the CLARITY Centre for Sensor Web Technologies under grant number 07/CE/I1147 and through the Insight Centre for Data Analytics under grant number SFI/12/RC/2289.
References

