Improving Recommendation Diversity

Keith Bradley [1] and Barry Smyth [1,2]

[1]Smart Media Institute,University College Dublin, Dublin, Ireland
[2] ChangingWorlds, Trintech Building, South County Business Park, Leopardstown,Dublin 18, Ireland
Keith.Bradley@ucd.ie, Barry.Smyth@ChangingWorlds.com

Abstract

Recommender systems offer users a more intelligent and personalised mechanism to seek out new information. Content-based recommender systems generally prefer to retrieve a set of items maximally similar to a users' query and/or profile. We argue that as new types of recommendation domains and tasks emerge, this blind faith in the similarity assumption begins to seem flawed. We show that very often recommendation diversity is important and that traditional recommendation systems are marred by poor diversity characteristics. We evaluate a new class of diversity-preserving algorithm capable of addressing this without compromising similarity or efficiency.

1 Introduction

Recommender systems ([1],[2]) have been heralded as potentially powerful solutions to the ubiquitous information overload problems that plague the citizens of our online world. Users are finding it more and more difficult to access the right information at the right time by conventional means, such as search engines, and this limits their ability to profit fully from the online revolution. Recommender systems try to solve this problem by offering a more intelligent and personalised way for users to seek out new information more quickly and easily.

Often a recommender will be designed to return a number of similar cases to provide a choice of recommendations. Travel or property recommenders typically return the k best cases (holiday packages or apartment listings) for a user query. The objective is to satisfy user needs with a single search and to maximise the likelihood of relevant cases appearing high up in the result list, hence the priority given to similarity.

We believe that this standard *pure* similarity-based retrieval strategy is flawed in some application domains. Consider a PC recommender: a user submits a query for a PC less than £1200, with modem and 128MB RAM. The top recommendation returned is for a Dell Dimension with 1.3GHz, 128MB RAM and a 15" screen. A good recommendation perhaps but what if the second, third and fourth recommendations are all slight modifications of the same model? All of these hit the mark with respect to the users profiled interests, but collectively provide poor coverage of the space of relevant recommendations for this user.

This scenario shows that while the k best recommendations are very similar to the target query or profile, they can also be very similar to each other. If the user decides to avoid Dell, then none of the alternative recommendations will suffice. By prioritising similarity during retrieval conventional recommenders implicitly ignore the importance of result diversity, and this may reduce the quality of the final recommendations. Recommenders should seek to provide optimal coverage of the information space in the vicinity of the users query.

This diversity problem is a recognised shortcoming of content-based recommendation techniques ([3],[4]). A common solution is to consider alternative recommendation techniques that are less susceptible to the diversity problem. For example, PTV operates in the TV listings domain, recommending TV programmes to users based on their learned viewing preferences [4]. PTV combines case-based recommendation with collaborative filtering in order to help guarantee a diverse set of recommendations. Similarly, CASPER [3], a job recommender system enhances a standard set of retrievals by using collaborative-filtering and client side personalisation.

In this paper we propose (Section 3) and evaluate (Section 4) a new type of diversity-preserving similarity-based retrieval algorithm capable of delivering significant improvements in recommendation diversity without compromising recommendation similarity. In particular we build on recent work [5] by focusing on different ways of combining similarity and diversity during recommendation. We begin by surveying some recent developments which, we believe, have profound implications for recommendation systems and further motivate the need to consider diversity and similarity in the recommendation process.

2 Recommendation Devices

Most work in the area of recommendation systems has focused on the development of systems designed to run on standard PC's. These systems implicitly assume the characteristics of this platform: large display areas; rich, multimedia capabilities; sophisticated input capabilities.

However, as new types of information access devices hit the consumer market new platforms will become the norm for the next-generation of recommendation systems. Indeed there is now sufficient evidence to suggest that shortly, there will be a shift away from PC-based Internet access towards devices such as WebTV, PDAs, and Internet-enabled mobile phones.

Reduced screen sizes and interface capabilities are probably the most significant factors affecting these new devices. For example the current generation of WAP-enabled phones suffer from screens that are up to 200 times smaller than a typical PC monitor. This severely reduces the number of recommendations that can be viewed in a single search. If all of the results are essentially the same, the chances of satisfying the user in a single search are greatly reduced. However, if the results are relevant *and* diverse then there is a much greater chance of success.

3 Combining Similarity and Diversity

In content-based recommenders, the normal approach to measuring the similarity between an information item c and target query t, is to use a weighted-sum metric (Equation 1). Selecting the k most similar items usually results in a characteristic similarity profile where average similarity of the result set reduces gradually for increasing values of k (see Section 4.2).

$$Similarity(t, c) = \frac{\sum_{i=1..n} w_i * sim_i(t_i, c_i)}{\sum_{i=1..n} w_i} \tag{1}$$

We define the diversity of a set of items, $c_1, ...c_n$, to be the average *dissimilarity* between all pairs of items in the result-set (Equation 2). Standard content-based recommenders often display a characteristic diversity profile with diversity increasing for larger result sets (see Section 4.2). Thus the trade-off between similarity and diversity is simple: for low values of k, while similarity tends to be high, diversity tends to be very low, highlighting the fundamental problem that exists with case-based recommenders.

$$Diversity(c_1, ...c_n) = \frac{\sum_{i=1..n} \sum_{j=i..n} (1 - Similarity(c_i, c_j))}{\frac{n}{2} * (n-1)} \tag{2}$$

In practice, improving the diversity characteristics of a fixed-size recommendation list means sacrificing similarity. Our goal is to develop a strategy that optimises this similarity-diversity trade-off, delivering recommendation sets that are diverse without compromising their similarity to the target query. We will describe three different strategies for retrieving k items from a data-base C, given a target query t, each focusing on a different way of increasing the diversity of the result set R.

```
t: target query, C: case-base, k: # results, b: bound
```

```
define BoundedRandomSelection (t, C, k, b)
begin
  C' := bk cases in C that are most similar to t
  R := k random cases from C'
   return R
end
```

```
define GreedySelection (t, C, k)
begin
  R := {}
  For i := 1 to k
    Sort C by Quality(t,c,R) for each c in C
    R := R + First(C)
    C := C - First(C)
  EndFor
return R
end
```

```
define BoundedGreedySelection (t, C, k, b)
begin
  C' := bk cases in C that are most similar to t
  R := {}
  For i := 1 to k
    Sort C' by Quality(t,c,R) for each c in C'
    R  := R + First(C')
    C' := C' - First(C')
  EndFor
return R
end
```

Figure 1: Diversity preserving algorithms

3.1 Diversity 1 - Bounded Random Selection

The simplest strategy for increasing the diversity of a set of k items is the Bounded Random Selection method *(Random)*: select the k cases at random from a larger set of the bk most similar cases to the target, with $b > 1$ (see Figure 1). Of course as $bk \rightarrow n$, Random becomes ineffective as a retrieval method since similarity is essentially ignored. Nevertheless this algorithm does serve as a benchmark against which to evaluate more principled strategies, and for lower values of b it will at least limit the similarity sacrifices albeit perhaps with only modest diversity improvements.

3.2 Quality Metrics

A more principled approach to improving diversity, while at the same time maintaining similarity, is to explicitly consider both diversity and similarity during retrieval. The remaining algorithms discussed here achieve this by considering the quality of individual items during retrieval, where quality is a measure that explicitly combines both similarity and diversity.

Many different types of quality metric are possible. For instance in earlier work [5] the very simple metric shown in Equation 3 was used: the quality of an item c is proportional to the similarity between c and the current target t, and to the diversity of c *relative* to those items so far selected, $R = \{r_1, ..., r_m\}$. The *relative diversity* metric shown in Equation 4 is a variation of the diversity metric from Equation 2.

$$Quality(t, c, R) = Similarity(t, c) * RelDiversity(c, R) \tag{3}$$

$$\begin{aligned} RelDiversity(c, R) &= 1 \; if \; R = \{\}; \\ &= \frac{\sum_{i=1..m}(1 - Similarity(c, r_i))}{m}, otherwise \end{aligned} \tag{4}$$

We will focus on a variation of this quality metric shown in Equation 5 in which the relative weight of the similarity and diversity factors can be changed by adjusting α.

$$Quality(t, c, R) = (1 - \alpha) * Similarity(t, c) + \alpha * RelDiversity(c, R) \tag{5}$$

Another alternative quality metric is shown in Equation 6. Here quality is computed to be the simple harmonic mean of similarity and diversity.

$$Quality(t, c, R) = 2/(\frac{1}{Similarity(t, c)} + \frac{1}{RelDiversity(c, R)}) \tag{6}$$

A wide variety of quality metrics could be examined, but are beyond the scope of this work. The need for a quality metric that is separate from core similarity metric is often questioned. It is argued that poor retrieval diversity is the result of a poor similarity metric. This is simply not true. Similarity and diversity are orthogonal measures. Similarity is a local function of two items, the target and a candidate, and the similarity of an item with respect to a target does not depend on the similarity of any other item. In contrast, the relative diversity of an item depends on previous similarity computations (and selections). For this reason it is not possible to fold diversity in to a single similarity computation - a separate quality metric is required.

3.3 Diversity 2 - Greedy Selection

The Greedy Selection algorithm *(Greedy)* in Figure 1 seeks to provide a more principled approach to improving diversity by using a quality metric to guide the construction of a result set, R, in an incremental fashion. During each iteration the remaining items are ordered according to their *quality* and the highest quality item added to R. The first item to be selected is always the one with the highest similarity to the target. During each subsequent iteration, the item selected is the one with the highest quality with respect to the set of cases selected during the previous iteration. As it stands this algorithm is expensive.

3.4 Diversity 3 - Bounded Greedy Selection

To reduce the complexity of the Greedy Selection algorithm we can implement a bounded version in the same spirit as the Bounded Random Selection algorithm. The Bounded Greedy Selection algorithm *(Bounded Greedy)*(see Figure 1) first selects the best bk cases according to their similarity to the target query (line 3) and then applies the greedy selection method to these (lines 4 - 11).

This new algorithm has a greatly reduced retrieval cost but since we are no longer examining all of the remaining cases we may miss a case with a marginally lower similarity value than the best bk cases but with a significantly better diversity value. Even if such a case has a higher overall quality value it will not be added to the retrieved set because it is not one of the best bk cases. However, the likelihood of this happening decreases with case similarity so that for suitable values of b it becomes unlikely [5].

4 Experimental Analysis

Our position is this: in many recommender system application scenarios, similarity and diversity *both* have roles to play in the recommendation process. However, trade-offs do exist. Increasing recommendation diversity means decreasing the similarity of the retrieved items to the target query. Our objective has been to develop an efficient strategy that is capable of maximally improving diversity while minimally compromising similarity.

In previous work [5] we have demonstrated that the Bounded Greedy algorithm is capable of maximally improving recommendation diversity without compromising similarity. In this section we describe a new series of experiments designed to evaluate the bounded greedy algorithm in a different domain, the recommendation of recruitment advertisements, and with a particular focus on the role of different quality metrics as part of the recommendation process.

4.1 Experimental Setup

Our data-base of recruitment advertisements is taken from one of Ireland's largest online recruitment services (www.jobfinder.ie). The data-base contains approximately 1000 different job adverts from the IT domain each one describing a particular job in terms of features such as *job title*, *location*, *experience*, *salary*, etc. We implement a number of recommender systems using the Bounded Greedy algorithm, but differing in the quality metric used:

- **BG(0.5),BG(0.75),BG(0.9)**: the Bounded Greedy Selection algorithm with $b = 2$ and the weighted quality metric with $\alpha = 0.5, 0.75$ and 0.9 respectively;

- **BG(Harm)**: the Bounded Greedy Selection algorithm with $b = 2$ and the simple harmonic mean version of quality.

We also implement a standard similarity-based technique (*OptSim*) using nearest-neighbour content-based recommendation. This delivers results with optimal similarity characteristics. In contrast, the *OptDiv* technique is a brute-force method of computing results with optimal diversity characteristics by examining all possible *k-element* result sets from the bk most similar items that are initially selected for a given target query. Finally, we implement a recommender system using the Bounded Random technique (*Random*).

4.2 Similarity Profiles

The most important issue is the trade-off between similarity and diversity in the various algorithms - is there an increase in diversity, and if so what is the similarity cost? To investigate this we run a series of retrievals using each recommender for various values of k. 100 items are chosen randomly from the data-base as queries, with the remaining items serving as the recommendation targets. For each recommendation session we measure the average similarity and diversity of the k results. This is repeated 20 times for different sets of 100 queries and we compute an overall average similarity and diversity value for each value of k and each recommender.

The similarity results are shown in Figure 2(A) as graphs of average similarity against k. The characteristic similarity profile is clearly seen for each recommendation strategy: in Figure 2(A) average similarity drops off with k. The OptSim strategy performs best across all values of k. For example, for $k = 5$ the average similarity for the OptSim strategy is approximately 0.92. This is compared to just under 0.9 for the OptDiv technique at $k = 5$, or just over 0.91 for the BG(0.5) variation. The BG(Harm) and BG(0.9) variations present with virtually identical similarity profiles. In this application domain there is a tendency to high similarity values during recommendation due to the high coverage characteristics of the recruitment case-base. Nevertheless the relative fall-off in average similarity is clear for the different recommendation strategies as k increases.

The Random technique performs relatively well as k increases. Although at $k = 3$ Random presents with the lowest average similarity, as k increases it beats many other recommendation strategies. At $k = 10$ it presents better similarity characteristics than all but the optimal

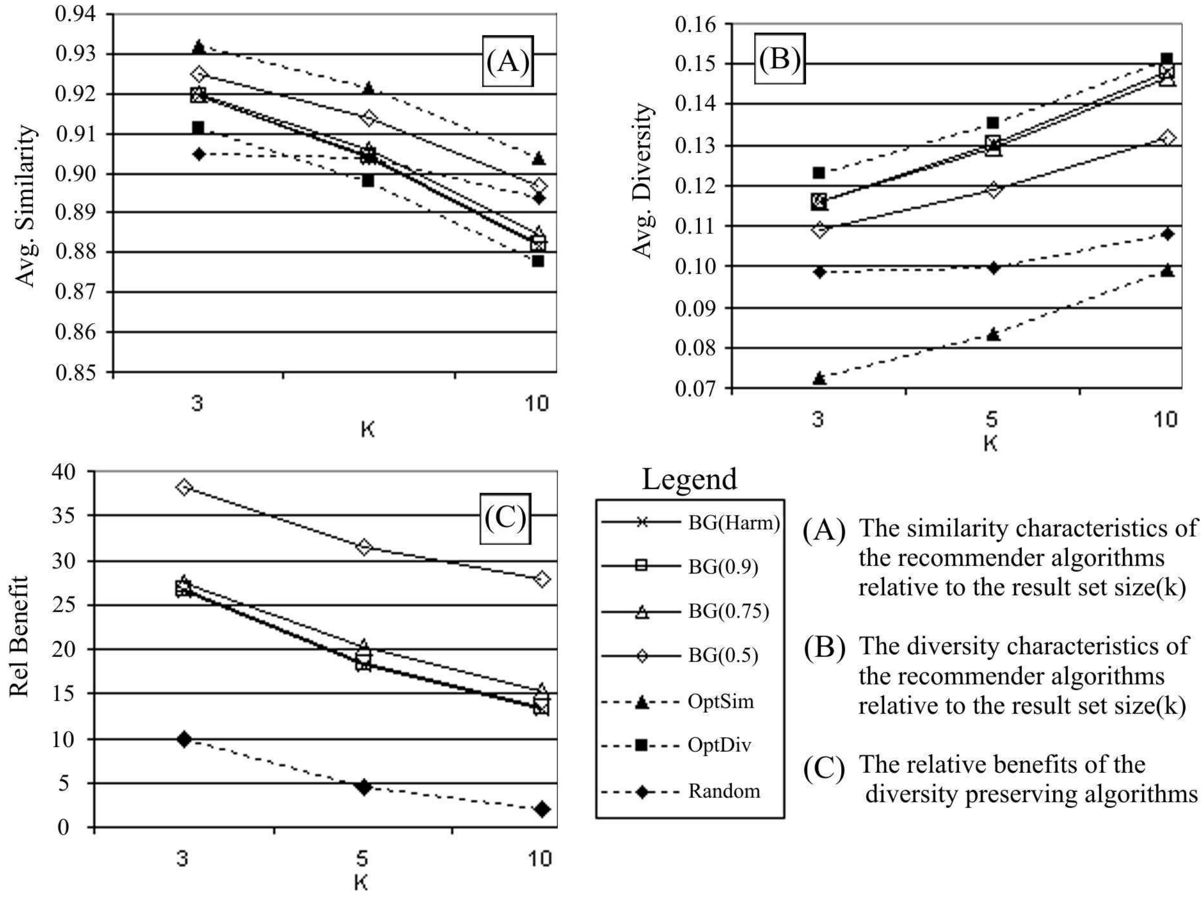

Figure 2: (A) similarity, (B) diversity and (C) relative benefits of the recommender algorithms

similarity technique and BG(0.5). This highlights the potentially significant similarity cost associated with the other (more aggressive) diversity-preserving strategies.

4.3 Diversity Profiles

The diversity results from the above experiment are shown in Figure 2(B) as graphs of average diversity versus k and the percentage diversity loss for each of the algorithms relative to OptDiv. As expected, the OptSim method performs poorly, delivering diversity values of between 0.07 and 0.1, compared to optimal diversity levels (OptDiv) of between 0.12 and 0.15. Clearly the bounded greedy diversity preserving algorithms perform much better than the standard OptSim approach used by many of today's content-based recommenders - all produce diversity values in excess of 0.11.

Looking at Figure 2(B) it is possible to appreciate the relative diversity gain for each of the diversity preserving algorithms, compared to the diversity levels of the OptSim method. The OptDiv method achieves diversity values between 150% and 170% greater than the diversity values of the result sets returned by OptSim. The bounded greedy algorithms achieve diversities between 130% and 159% greater than the OptSim levels. This means that the OptSim results have diversity levels that are approximately 60% of the optimal diversity levels, whereas the bounded greedy techniques are capable of selecting results with well over 90% of optimal diversity.

4.4 Relative Benefits

The results highlight how different strategies trade-off similarity losses for diversity gains, but the precise trade-offs between similarity and diversity are not yet clear. We previously examined this issue by comparing similarity losses to diversity gains [5] but the point has been made that it may not be reasonable to compare such values directly. We propose to look instead at how each strategy results in a percentage loss in similarity relative to the optimal similarity strategy, and a percentage gain in diversity relative to the optimal similarity strategy.

In Equation (7) the Relative Benefit *(RBen)* of using a retrieval algorithm is equal to the increase in diversity gained by using the algorithm relative to the diversity of *OptSim* divided by the decrease in average similarity of the recommendations relative to the similarity of *OptSim*. The similarity and diversity measures are taken as a percentage of *OptSim* and *OptDiv* respectively so as to normalise the results.

$$RBen(R) = \left(\frac{Diversity(R) - Diversity(OptSim)}{Diversity(OptDiv)}\right)/\left(\frac{AvgSim(OptSim) - AvgSim(R)}{AvgSim(OptSim)}\right) \quad (7)$$

$$where Avg.Sim(X) = \frac{\sum_{i=1..n} Similarity(t, x_i)}{n} \quad (8)$$

At the $k = 5$ level the BG(0.5) algorithm result sets have an average similarity of 0.913, put another way that is over 99% of the average optimal similarity and an average diversity of 0.118, i.e. 87% of the average optimal diversity. The results obtained using *OptSim* gives an average similarity of 0.921 which naturally is 100% of the average optimal similarity, but only gives a diversity of 0.083 or approximately 61% of the average optimal diversity. In other words, the BG(0.5) strategy suffers from less than a 1% loss in similarity but benefits from an approximately 26% gain in diversity, relative to the optimal similarity recommendation strategy; i.e. a relative benefit of 31 (26.05% gain in diversity / 0.8% loss in similarity). Thus a relative benefit of 1 would mean that a recommendation strategy suffers from a percentage loss in similarity which is exactly matched by its percentage gain in diversity.

The results are shown in Figure 2(C) as a graph of relative benefit against k. As expected the Random strategy performs poorly with the lowest relative benefit values recorded across all values of k. In general the BG(0.5) strategy performs best, delivering relative benefit values of between 27 and 38 for the different values of k, which places it significantly higher than all other strategies. The remaining strategies, BG(0.9), BG(0.75), and BG(Harm) all present with intermediate relative benefit values.

We also find that relative benefit values tend to fall off with increasing k for all strategies. This suggests that the maximum benefits are to be found for a low values of k, particularly important in the context of mobile recommender systems, where limited screen spaces restricts the number of recommendations that may be returned.

5 Conclusions

We have argued that the traditional recommender system strategy, whereby the most similar cases are retrieved for a given target query, suffers from potentially poor diversity characteristics, which may limit the usefulness of future recommender systems. We have argued for the need to explicitly consider diversity during the recommendation process, and presented and evaluated a variety of diversity preserving algorithms. We have extended our work in this area to focus on how varying the influence of similarity and diversity, during the recommendation process, impacts on the quality of the final result sets. In addition we have applied our techniques in the new domain of online recruitment. These experiments have shown that significant diversity gains are available without compromising similarity. In general our findings highlight the benefit of being able to adjust the weighting associated with similarity and diversity, and that an over emphasis on diversity will result in a corresponding drop in similarity, and lower relative benefits. In the test domain, the BG(0.5) strategy delivered the best overall results with superior relative benefits and minimal loss in similarity.

References

[1] Burke, R.: A case-based approach to collaborative filtering. In: *Proceedings of the 5th European Workshop on Case-Based Reasoning*. Springer-Verlag, 2000.

[2] Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gorgan, L.R., and J. Riedl.: GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3):77–87, 1997.

[3] Rafter, R., Bradley, K., and Smyth, B.: Personalised Retrieval for Online Recruitment Services. In: *Proceedings of the 22nd Annual Colloquium on Information Retrieval. Cambridge, UK.*,2000

[4] Smyth, B. and Cotter, P.: A Personalized TV Listings Service for the Digital TV Age. *Journal of Knowledge-Based Systems*, 13(2-3):53–59, 2000.

[5] Smyth, B.: and McClave, P.: Similarity vs Diversity. In: *Proceedings of the 4th International Conference on Case-Based Reasoning* Springer-Verlag, 2001.