On the Role of Diversity in Conversational Recommender Systems

Lorraine McGinty[1] and Barry Smyth[1,2]

[1] Smart Media Institute, University College Dublin, Dublin, Ireland
[2] ChangingWorlds, University College Dublin Dublin, Ireland
Barry.Smyth@ChangingWorlds.com

Abstract. In the past conversational recommender systems have adopted a similarity-based approach to recommendation, preferring cases that are similar to some user query or profile. Recent research, however, has indicated the importance of diversity as an additional selection constraint. In this paper we attempt to clarify the role of diversity in conversational recommender systems, highlighting the pitfalls of naively incorporating current diversity-enhancing techniques into existing recommender systems. Moreover, we describe and fully evaluate a powerful new diversity-enhancing technique that has the ability to significantly improve the performance of conversational recommender systems across the board.

1 Introduction

Conversational recommender systems are forwarded as a potential solution to the information overload problem that are particularly well-adapted to consumer-oriented e-commerce applications. They help users to navigate through complex product spaces in pursuit of suitable products by presenting sets of alternative recommendations (during a series of *recommendation cycles*) and taking advantage of user feedback to guide future cycles.

Usually, conversational recommenders adopt a similarity-based recommendation strategy, selecting cases for recommendation because they are are maximally similar to the current user query. However, recently this emphasis on similarity as the primary selection pressure has been brought into question. In particular, the diversity of a set of recommendations has been shown to be important, but largely independent of query similarity. A set of recommendations may be similar to the current query but if they are also very similar to each other - if they lack diversity - then these recommendations provide only limited coverage of the recommendation space. In response a number of researchers have proposed techniques for improving the diversity of a set of recommendations without significantly (if at all) affecting similarity to the user query. However, although such techniques make it possible to improve the diversity of a set of recommendations, no attempt has been made to consider how and when such methods should be used properly in conversational recommender systems.

The assumption that such methods can be used to increase the diversity of each set of recommendations ([10]) is well made. However, the degree to which diversity can be used to improve the efficiency and/or quality of a recommender system has not yet been evaluated. While it has been shown that diversity can be increased within a single recommendation cycle, the degree to which this can help a conversational recommender more efficiently focus in on a good recommendation for the target user, is still unclear. Indeed employing current diversity-enhancing techniques during each recommendation cycle of a conversational recommender is unlikely to deliver optimal results; these techniques are designed to pass-over items that are similar to the target query if they are also similar to concurrent recommendations. Obviously, if the target item is one of these passed-over items, then an opportunity to successfully complete the recommendation task in the current cycle will have been missed.

In this paper we investigate this issue empirically on a range of data-sets, highlighting inefficiencies with current diversity-enhancing techniques. We also present a novel approach to controlling diversity in conversational recommenders called *adaptive selection* and demonstrate that it can lead to significant performance improvements in recommenders that employ preference-based [3–5] and critiquing [2] as their primary forms of feedback.

2 Similarity vs Diversity in Conversational Recommender Systems

Conversational recommender systems engage the user in an extended dialog to better ascertain a user's specific requirements and needs in order to make relevant recommendations. Shimazu [10] distinguishes between two basic approaches. With *Navigation by asking* the user is engaged in a question-answering dialog where they are asked to answer specific questions about the features of their ideal product case. For example, in a PC recommender the user may be asked to provide information about the processor, memory or pricing requirements. In contrast, with *navigation by proposing* the user is instead presented with suggested recommendations and asked to provide feedback regarding the suitability of these suggestions. In particular, the common forms of feedback used in navigation by proposing are preference-based feedback and critiquing. For example, in a PC recommender that uses preference-based feedback a user would simply indicate a preference for PC1 over PC2 or PC3. In a recommender using critiquing the user would indicate such a preference but also provide a critique. For example they might indicate that they are looking for *less expensive* cases than PC1 by critiquing the *price* feature.

Traditionally, conversational recommender systems have followed a similarity-based recommendation policy irrespective of their basic recommendation approach or the type of feedback that they rely on. Accordingly k cases are selected for recommendation because they are maximally similar to the current user query; in the terminology of [9] this set of k cases is called the *standard retrieval set* or *SRS*. However, one of the more significant recent developments has been the growing realisation that similarity on its own may not be an ideal selection constraint. Smyth & McClave [11] highlighted how similarity-based approaches had a tendancy to produce recommendations that lacked diversity and the problems that this entails. The recommended cases may all be similar to the current query but if they are also very similar to each other then their value as a set of alternatives for the user to judge will be compromised. This observation motivated the need for new selection methods capable of delivering a set of recommendations that are diverse as well as being similar to the user query.

The *bounded greedy* technique introduced by [11] was the first attempt to explicitly enhance the diversity of a set of recommendations without significantly compromising their query similarity characteristics; although it is worth noting that some loss of similarity is experienced with this approach. Both methods operate in a similar 2-pass manner. A first pass over the recommendable items (case-base) rank cases according to their query similarity. The second pass sequentially transfers cases from this ranked list to the final recommendation list such that at each transfer point the case that is selected is the one maximises the product of its similarity to the target query and its diversity relative to the cases that have already been selected; the diversity of a case c relative to a set of cases C is given by Equation 1.

$$RelDiv(c, C) = \frac{\sum_{i=1}^{n}(1 - Sim(c, C_i))}{n} \tag{1}$$

In parallel Shimazu [10] introduced an alternative method for enhancing the similarity of a set of recommendations. In brief, a set of 3 recommendations, c_1, c_2 and c_3, are chosen relative to some query q such that c_1 is maximally similar to q, c_2 is maximally dissimilar to c_1 and then c_3 is maximally dissimilar to c_1 and c_2. In this way, the triple of cases are chosen to be maximally diverse but unlike the bounded greedy technique above the similarity of c_2 and c_3 to the query is likely to be compromised. As such the value of this approach is limited to situations where the set of recommended cases is drawn from a set of cases that are all sufficiently similar to the user query to begin with.

Recently a number of alternative diversity enhancing selection techniques have been proposed. For example, [7] shows that it is sometimes possible to enhance diversity without loss of query similarity. An approach to enhancing diversity based on the idea of *similarity layers* is described [7, 9]. Very briefly, a set of cases, ranked by their similarity to the target query, can be partitioned into similarity layers, such that all cases in a given layer have the same similarity value to the query. To select a set of k diverse cases, the lowest similarity layer that contributes cases to the SRS is identified and a subset of cases from this layer are selected for inclusion in the final recommended set; all cases in higher similarity layers are automatically included. Cases are selected from this lowest similarity layer using an optimal diversity maximizing algorithm. This approach has the ability to improve diversity while at the same time fully preserving the similarity of cases to the user query. However, the diversity improvements obtained are typically less than those achieved by the bounded greedy algorithm.

An alternative, and more flexible, diversity enhancing approach is also introduced based on the analogous notion of *similarity intervals* and the selection of cases from the *rightmost* similarity interval that contributes cases to the SRS; all cases in a given similarity interval are within a specific similarity of the query. The advantage of this approach is that it can achieve greater diversity improvements by relaxing the constraint that query similarity must be preserved. Query similarity is reduced but within a tolerance level defined by α, the width of the similarity intervals.

It is also worth noting that a retrieval technique may not be designed to explicitly enhance diversity but may nonetheless have a beneficial effect by its very nature. *Order-based retrieval* is a good example of such a technique [1]. It is based on the idea that the relative similarities of cases to a query of *ideal* feature values is one way of ordering a set of cases for recommendation. Very briefly, order-based retrieval constructs an ordering relation from the query provided by the user and applies this relation to the case-base returning the k cases at the top of the ordering. The order relation is constructed from the composition of a set of canonical operators for constructing partial orders based on the feature types that make up the user query. The essential point to note is that an empirical evaluation of order-based retrieval demonstrates that it has an inherent ability to enhance the diversity of a set of retrieval results; that is, the cases at the top of the ordering tend to be more diverse than an equivalent set of cases ranked based on their pure similarity to the user query.

All of the above techniques have been shown to improve the diversity of a single set of recommendations while preserving the similarity of these recommendations to the query to a lesser or greater extent. In other words, using the above techniques it is possible to increase the diversity of a given recommendation set in a given recommendation cycle. However, there has been no attempt to assess the implications of such diversity enhancing methods across the multiple recommendation cycles that make up the dialog of a conversational recommender system. In particular, most of the above techniques operate by eliminating certain cases from the recommended set: cases that would otherwise have been selected on the basis of their similarity to the user query; cases that are not diverse relative to others that have been selected. If one of these cases happens to be the ideal target case for the user, or a case that may lead more directly to the ideal target, then the efficiency of the conversational recommender may be compromised.

3 Diversity Enhancement in Comparison-Based Recommendation

Comparison-based recommendation is a generic framework for conversational recommender systems that emphasises the roles of case selection, user feedback, and query modification during navigation by proposing. Although initially comparison-based recommendation was proposed as a framework for investigating similarity-based recommenders utilising pure preference-based feedback, it is in fact sufficiently generic to accommodate a range of different recommendation strategies and feedback types. We will describe how it can be adapted to incorporate diversity enhancing selection techniques such as those described above. Crucially, however, we will also describe an important new technique that introduces diversity into the recommendation process in a more selective way by judging whether it is appropriate or not for a given recommendation cycle.

3.1 Comparison-Based Recommendation

The basic comparison-based recommendation algorithm (Figure 1) describes an iterative process that starts with an initial user query and terminates when the user is satisfied with one of the recommendations proposed. During each iteration, or recommendation cycle, a set of k cases are chosen according to some selection mechanism, feedback is provided by the user based on their preference for a given case, and the current query is updated based on this feedback.

```
define Comparison-Based-Recommend(q, CB, k)
begin
  do
    R ← ItemRecommend(q, CB, k)
    i_p ← UserReview(R, CB)
    Q ← QueryRevise(q, i_p, R)
  until UserAccepts(i_p)
end

define ItemRecommend(q, CB, k)
begin
  CB' ← sort cases in CB in decreasing order of their sim to q
  R ← top k items in CB'
  return R
end

define UserReview(R, CB)
begin
  i_p ← user's preferred case from R
  CB ← CB - R
  return i_p
end

define QueryRevise(q, i_p, R)
begin
  R' ← R - {i_p}
  q ← i_p
  return q
end
```

Fig. 1. The comparison-based recommendation algorithm using simple similarity-based retrieval with preference-based feedback.

```
define ItemRecommend(q, CB, c, k)
begin
  CB' ← {i ∈ CB | Satisfies(i,c)}
  CB'' ← sort CB' in by decreasing sim to q
  R ← top k items in CB''
  return R
end

define UserReview(R, CB)
begin
  i_p ← user's preferred case from R
  c ← user critique for some f ∈ i_p
  CB ← CB - R
  return <i_p,c>
end
```

Fig. 2. Modifications to the comparison-based recommendation algorithm when critiquing is the form of feedback used.

As it stands the algorithm shown in Figure 1 utilises simple similarity-based recommendation with preference-based feedback, in which the user indicates a simple preference for one of the k recommended cases. Moreover, on the basis of this feedback the user query is replaced with the preferred case for the next recommendation cycle; in previous work we have described and evaluated a range of more sophisticated query revision strategies [3, 4]. However, we have also shown previously how to adapt comparison-based recommendation to use critiquing [12] as a form of feedback and these modifications are shown in Figure 2.

3.2 Uniform Diversity Enhancement

It is straightforward to adapt the similarity-based comparison-based recommendation methods described above to incorporate diversity-enhancement techniques by updating the *ItemRecommend* function. For example, in Figure 3, we show how the bounded greedy method [11] can be used to update this function when preference-based feedback is used; a similar modification can be made for critiquing. Using this modification we can implement conversational recommender systems that use preference-based or critiquing as a form of user feedback, and that give consideration to query similarity and recommendation diversity during each recommendation cycle. It is vital to note that these adaptations introduce diversity in a uniform manner into each and every recommendation cycle.

3.3 Selective Diversity Enhancement

The value of increasing the diversity of the cases in a given recommendation cycle is that it allows for a broader set of alternatives to be presented to the end user. This is likely to be useful at certain stages during the recommendation dialog.

```
define ItemRecommend(q, CB, k)
begin
  α = 0.5
  R ← BoundedGreedySelection (q, CB, k, b, α)
  return R
end

define BoundedGreedySelection (q, CB, k, b, α)
begin
  CB' := bk items in CB that are most similar to q
  R := {}
  For j := 1 to k
    Sort CB' by Quality(q,i,R) for each case i in CB'
    R  := R + First(CB')
    CB’ := CB’ - First(CB')
  endFor
  return R
end

… where Quality(q,i,R) = α.Sim(q,i)+(1- α)RelDiv(i,R)
        RelDiv(i,R)= ∑(j=1..m) (1-Sim(i,r_j))/m if R={}; = 1 otherwise
```

Fig. 3. Adapting the similarity-based comparison-based recommendation algorithm to incorporate diversity-enhancement techniques.

For example, if the users requirements are unclear then presenting a diverse set of cases allows the recommender to cover a number of different points in the recommendation space in the hope that one will be a fair match for the user’s needs. Indeed similar techniques are used by sales assistants in real-world sales dialogs. When a customers needs are unclear the sales assistant will present a diverse set of options to try and focus in on a particular type of product. However, as the users needs become more refined the sales assistant will tend to switch their recommendation strategy, suggesting products that are as similar as possible to the users known requirements as they home in on the right region of the product space.

These observations motivate the need for a more sophisticated recommendation strategy, one that adapts its use of similarity and diversity depending on whether the recommender has focused in on the right region of the recommendation space. If the current recommendation focus appears to be correct then a similarity-based strategy is appropriate in order to refine the recommendations in the region of the target item. If the focus appears to be incorrect then recommendation diversity can be increased in order to broaden the search. The trick is how to determine whether or not the recommender is correctly focused.

Adaptive Selection is an attempt to produce a more sophisticated recommendation strategy that is capable of adjusting the balance of similarity and diversity during each recommendation cycle. It takes advantage of the key idea that it is possible to determine whether or not the recommender is correctly focused by determining whether the recent recommendations represent an improvement on those made in the previous cycle. This is achieved by making two further modifications to the basic comparison-based recommendation technique. First, instead of making k new recommendations in each new cycle, the current

```
define ItemRecommend(q, CB, k, i_p, i_p-1)
begin
  if(i_p != null) && (i_p == i_p-1)
     R` ← ReFocus(q, CB, k-1)
  else
     R` ← ReFine(q, CB, k-1)
  R ← R`+ i_p
  return R
end

define ReFine(q, CB, k)
begin
  CB' ← sort CB in decreasing order of their sim to q
  R ← top k items in CB'
  return R
end

define ReFocus(q, CB, k, i_p, i_p-1)
begin
  α = 0.5
  CB' ← sort CB in decreasing order acc to Equation 2
  return BoundedGreedySelection(q, CB, k, b, α)
end
```

Fig. 4. Adapting the similarity-based comparison-based recommendation algorithm for the 'adaptive selection' technique with preference-based feedback.

preference case (or the critiqued case) is added to $k-1$ new recommendations; we refer to this as *carrying the preference* (CP). On its own this modification introduces redundancy, in the sense that a previously seen case is repeated in one or more future cycles. However, including the previous preference makes it possible to avoid the problems that ordinarily occur when none of the newly recommended cases are relevant to the user; the user can simply reselect the carried preference case instead of being forced to follow a less relevant recommendation.

The essential point is that if the user prefers (or critiques) a case other than the carried preference, then it must be because it is closer to the target, and thus positive progress has been made. In this situation diversity is not warranted and the emphasis should be on similarity in the next recommendation cycle. If, however, the user prefers the carried preference case then it suggests that the other $k-1$ cases are less relevant than the carried case, and thus that the recommender has failed to make positive progress towards the target. In this situation two things happen. First, diversity is introduced into the next recommendation cycle. And secondly, during the selection of the new cases for the next recommendation cycle, the dissimilarity of these candidate cases to the rejected cases is taken into account. The basic idea is to prioritise cases that are not only similar to the query, but also dissimilar to the rejected cases. This is achieved by using the formula given below in Equation 2, where c is a candidate case, c_p is the current preferred case, and C' is the set of $k-1$ rejected cases.

$$SimDissim(c, c_p, C') = \frac{Sim(c, c_p) + \sum_{\forall c_i \epsilon C'} (1 - Sim(c, c_i))}{k} \qquad (2)$$

The algorithm components in Figure 4 are the modifications needed to implement adaptive selection for use in comparison-based recommendation with preference-based feedback. Once again, directly analogous modifications can be made for comparison-based recommendation when critiquing is the form of feedback used. The basic change is that the *ItemRecommend* function must first check whether the carried preference case has been selected by the user as their preference. If it has then the *Refocus* function is called to select a set of $k - 1$ diverse cases for the next cycle; they will be added to the preference case to make up the k cases for the next cycle. If, on the other hand, the carried preference has not been selected then the *Refine* function is called to help the recommender home in on the region of this recent preference in the hope that this region is occupied by the ideal target case.

4 Evaluation

So far we have argued for the potential advantages of incorporating diversity enhancing selection techniques into conversational recommender systems. However, we have also argued against the naive use of such techniques - where diversity is enhanced in each and every recommender cycle - pointing out that such strategies are likely to lead to protracted recommendation dialogs in some circumstances. In response we have forwarded the adaptive selection technique as a more selective use of diversity that is likely to improve recommendation efficiency. We test these claims in this section by empirically evaluating performance of similarity-based and diversity-enhanced recommendation techniques on a number of standard data-sets.

4.1 Setup

Algorithms We wish to test three basic conversational recommendation strategies: (1) SIM - a pure similarity-based recommender that serves as a benchmark; (2) DIV - a recommender that adopts the uniform diversity enhancing technique described above; and (3) AS - an equivalent recommender that adopts the adaptive selection technique. In addition, we wish to test these recommender systems using two different types of user feedback: preference-based feedback and critiquing. This gives six different recommender systems to test - SIM, DIV, and AS with preference-based feedback and SIM, DIV and AS with critiquing - each implemented using the comparison-based recommendation framework with $k = 3$.

Data-Sets The familiar *Travel* case-base contains 1024 cases, each describing a specific vacation in terms of features such as *location*, *duration*, *accommodation*, *price* etc. The *Whiskey* case-base ([6]) contains a set of 552 cases, each describing a particular Scotch whiskey in terms of features such as *distillery*, *age*, *proof*, *sweetness*, *flavour*, *finish* etc.

Methodology Using a leave-one-out methodology, each case (*base*) in a case-base is temporarily removed and used in two ways. First it serves as the basis for a set of queries constructed by taking random subsets of item features. Second, we select the case that is most similar to the original base. These cases serve as the recommendation *targets* for the experiments. Thus, the base represents the ideal query for a user, the generated query is the initial query that the user provides to the recommender, and the target is the best available case for the user based on their ideal. Each generated query is a test problem for the recommender, and in each recommendation cycle the users preference is assumed to be the case that is most similar to the known target case. Preference-based or critiquing is applied to this preference case as appropriate; in the case of the latter, a random critique is applied to the preferred case in each cycle.

Test Queries For each data set, three different groups of queries are generated of varying degrees of difficulty (*easy*, *moderate*, *difficult*); difficulty is based on the number of cycles required by STD.

4.2 Recommendation Efficiency

Perhaps the most basic test of a conversational recommender system concerns its recommendation efficiency [8]; that is, the length of the recommendation dialog for a typical query. Dialog length can be measured in terms of the number of cycles or the number of unique cases presented during the dialog. As such, to test recommendation efficiency the leave-one-out method outlined above is used for each query from both data sets across the three recommenders and the average number of cycles and unique items presented to the user are measured.

Results The results for Travel and Whiskey are summarised in Figures 5 and 6; in each case graph (a) and (b) relate to preference-based feedback and (c)and (d) relate to critiquing. Graphs (a) and (c) measure efficiency in terms of unique cases - the cycles data has been omitted for space reasons - but graphs (b) and (d) include both unique cases and cycles information in terms of the percentage benefit enjoyed by DIV and AS methods relative to the SIM benchmark.

Analysis A number of issues are clarified by these results. Firstly, a significant efficiency benefit is enjoyed by DIV and AS, when compared to SIM, in both data-sets, across all query types, and for both types of feedback. For example, using preference-based feedback in Travel, for moderate queries, the SIM method presents the user with an average of about 100 unique cases before the target case is located. By comparison, the DIV method requires only about 53 cases and the AS method requires only 24 cases. Using critiquing in Travel, the equivalent results for SIM, DIV and AS are 40, 35 and 19, respectively. The corresponding relative benefits are shown in Figure 5(c) for Travel with DIV experiencing a reduction in unique cases of 47% with preference-based feedback but only 13%

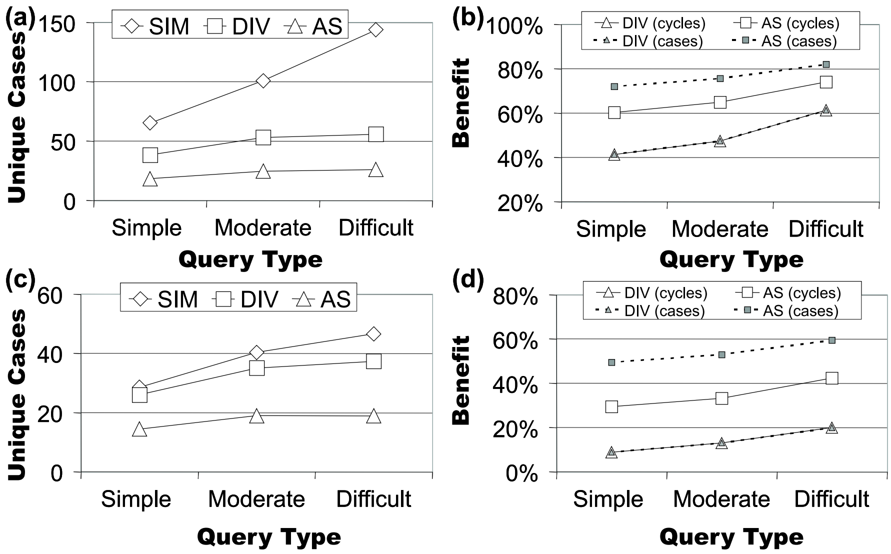

Fig. 5. Efficiency results for the *Travel* dataset using preference-based feedback (a & b) and critiquing (c & d). Graphs (a) and (c) measure efficiency in terms of unique cases. Graphs (b) and (d) include both unique cases and cycles information in terms of the percentage benefit enjoyed by DIV and AS methods relative to the SIM benchmark.

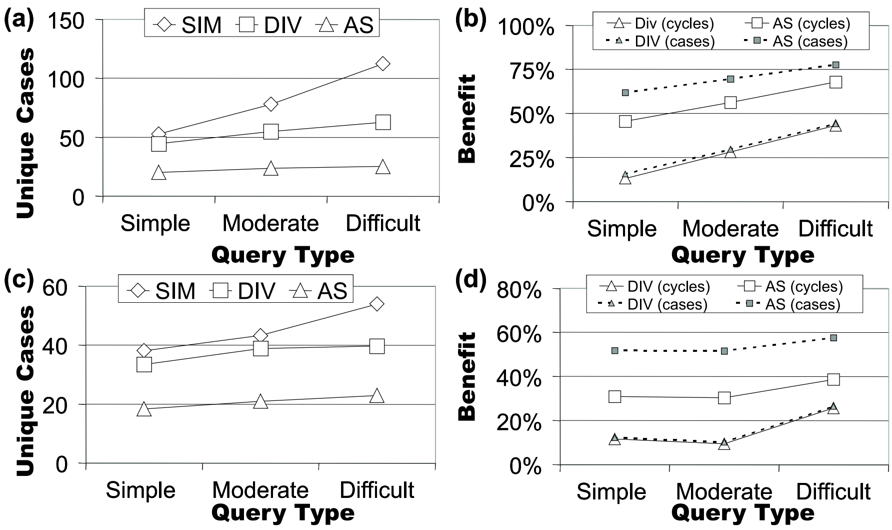

Fig. 6. Efficiency results for the *Whiskey* dataset. Graphs (a-d) represent the results found for the same comparisons as listed above.

for critiquing, relative to SIM. However, AS experiences comparable reductions of 76% and 53% relative to SIM; similar benefits are presented in terms of the reduction in cycles. Comparable results are also presented in Figures 5(a-c) for the Whiskey data-set. These results also validate our hypothesis that the uniform use of diversity in conversational recommenders is likely to be suboptimal: the efficiency of DIV is significantly less than that of AS across the board. Indeed, in both data-sets, the recommenders that employ critiquing enjoy only a limited efficiency improvment (<26%) when using the DIV method, compared to SIM. In contrast, the more sophisticated AS method displays a much greater efficiency advantage, achieving maximum efficiency improvments in excess of 75% (in terms of unique cases) in both data-sets for preference-based feedback, and up to 60% in both data-sets for critiquing. Finally, it is also worth highlighting how the efficiency benefits enjoyed by DIV and AS, relative to SIM, are generally increasing with query difficulty in both data-sets and using both types of feedback.

4.3 Preference Tolerance

The above assumes that the recommendation dialog ends when the pre-determined target case is selected by the user. This is analogous to a user seeking out a very specific case. In reality users are likely to be more flexible in their acceptance criteria, often tolerating cases that are close to, but not an exact match for, the target. To test this we repeat the above experiment but instead of terminating a recommendation dialog when the target has been found, it is terminated once a case is found that is within a specific similarity threshold of the target. We test similarity thresholds from 60% to 100%; 100% corresponds to the previous setup where the dialog terminates with the optimal target case.

Results The results for Travel and Whiskey are summarised in Figures 7 and 8; in each figure, graphs (a) and (b) relate to preference-based feedback, while (c) and (d) relate to critiquing. Also the graphs only present the results for queries of moderate difficulty; the results for the simple and advanced queries are broadly similar but omitted, once again, for space-saving reasons.

Analysis Once again the results are clear. The performance advantages enjoyed by DIV and by AS, relative to SIM, are once again found under less stringent success conditions. For example, in Travel with preference-based feedback, for moderate queries, we find that, on average, SIM expects users to look at about 61 unique cases at the 60% similarity threshold (compared to 100 cases at the 100% threshold). In comparison, under the same conditions, DIV and AS require the user to examine only 31 and 18 cases, respectively, representing a reduction of about 49% for DIV and 71% for AS, relative to SIM (see Figure 7(b)). As the similarity threshold increases, so too does the number of unique cases that the user must examine before locating a satisfactory one. Interestingly, while the relative benefit enjoyed by DIV remains relatively constant for changing

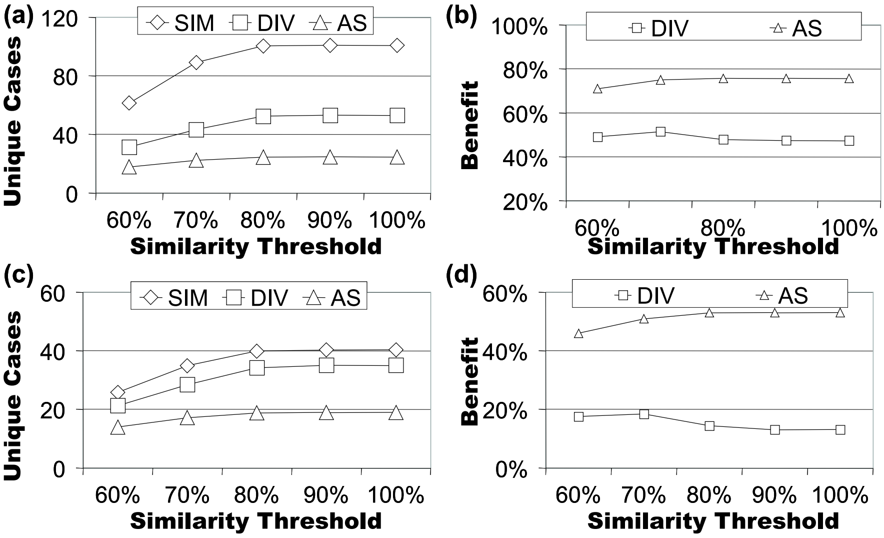

Fig. 7. Preference tolerance results for the *Travel* dataset. Graphs (a) and (b) relate to preference-based feedback, while (c) and (d) relate to critiquing. Graphs (b) and (d) summarise the percentage benefit (in terms of unique cases information) enjoyed by DIV and AS methods relative to the SIM benchmark.

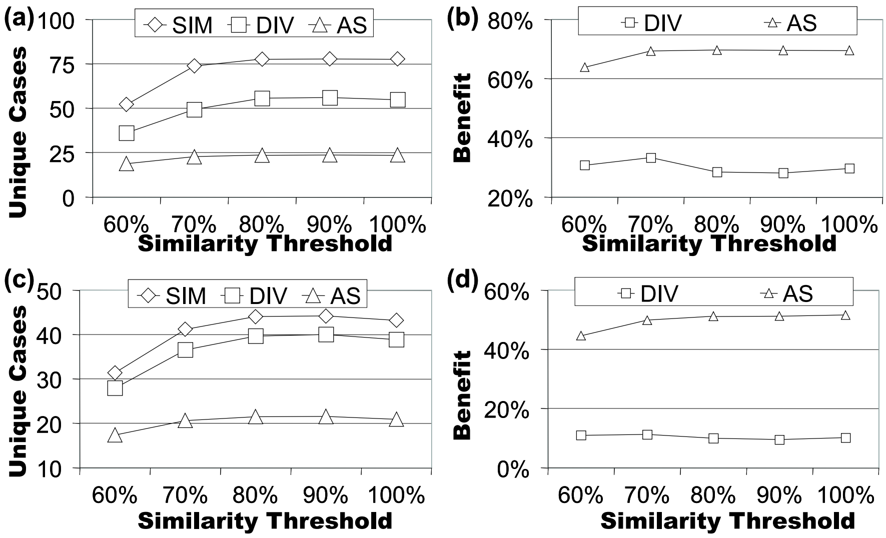

Fig. 8. Preference tolerance results for the *Whiskey* dataset. Graphs (a-d) represent the results described above for this dataset.

similarity thresholds, we find that the AS benefit is actually increasing with the similarity threshold. In other words, the ability of AS to improve upon the efficiency of SIM (or indeed DIV) increases as the success criterion becomes more stringent. These results for Travel with preference-based feedback are mirrored by equivalent results for Travel with critiquing. Likewise, the Whiskey results show a similar pattern (Figure 8(a-c)).

4.4 Conclusions

In this paper we set out to examine the role of diversity as part of the selection mechanism used in various kinds of conversational recommender systems. We highlighted that although some recent research had motivated the use of diversity as part of the recommendation process, so far only very limited evaluation work had been conducted, particularly when it came to testing the value of diversity across the many cycles of a typical conversational recommender system.

In this paper we have demonstrated that, in general there is a significant advantage to be gained from introducing diversity into each cycle of the recommendation process. Experiments with a number of complex case-bases show that even a simple uniform approach to introducing diversity has the potential to significantly enhance the efficiency of a conversational recommender system that utilises either preference-based or critiquing forms of feedback. For example, introducing diversity has the potential to reduce the number of unique cases that a user must examine by up to 60% in the case of preference-based feedback and by up to 20% in the case of critiquing.

However, the key contribution of this work is the idea that, powerful and valuable as diversity may be as a key selection constraint, it may not be warranted during every recommendation cycle. Indeed, increasing diversity may result in a loss in recommendation efficiency if the required case is eliminated from the recommended set by the diversity constraint. Accordingly, we have described a novel approach to case selection, called adaptive selection. Our approach takes advantage of similarity and diversity in a more intelligent manner, by determining when best to increase selection diversity and when best to focus on query similarity. Moreover, we have shown that this approach enjoys significant performance improvements over standard similarity-based recommenders and the more traditional (uniform) diversity enhancing techniques mentioned above. For example, adaptive selection can reduce the number of unique cases that a user must examine by up to 80% in the case of preference-based feedback and by up to 60% in the case of critiquing.

To conclude then: we have clarified the role of diversity as a primary selection constraint in conversational recommender systems. Diversity is an important selection constraint. And it can be used to more efficiently guide the recommendation process in many different varieties of conversation recommender systems, especially those that employ preference-based or critiquing as their primary forms of feedback. However, its proper use must be fine-tuned according to the progress being made by the recommender. If the recommender system appears to be close to the target case then diversity should be limited to avoid missing this case. But if the recommender system is not correctly focused on the right region of the recommendation space then diversity can be used to help refocus the recommender on the target region more quickly. The adaptive selection technique is capable of fine-tuning the use of diversity in this way and has the potential to have a major impact on the development of more efficient conversation recommender systems going forward.

References

1. D. Bridge. Diverse Product Recommendations using an Expressive Language for Case Retreival. In S. Craw and A. Preece, editors, *Proceedings of the Sixth European Conference on Case-Based Reasoning (ECCBR 02)*, pages 42–57. Springer, 2002. Aberdeen, Scotland.
2. R. Burke. Knowledge-based Recommender Systems. *Encyclopedia of Library and Information Systems*, 69(32), 2000.
3. L. McGinty and B. Smyth. Comparison-Based Recommendation. In S. Craw and A. Preece, editors, *Proceedings of the Sixth European Conference on Case-Based Reasoning (ECCBR 2002)*, pages 575–589. Springer, 2002. Aberdeen, Scotland.
4. L. McGinty and B. Smyth. Deep Dialogue vs Casual Conversation in Recommender Systems. In F. Ricci and B. Smyth, editors, *Proceedings of the Workshop on Personalization in eCommerce at the Second International Conference on Adaptive Hypermedia and Web-Based Systems (AH 2002)*, pages 80–89. Springer, 2002. Universidad de Malaga, Malaga, Spain.
5. L. McGinty and B. Smyth. Evaluating Preference-Based Feedback in Recommender Systems. In R. Sutcliffe, editor, *Proceedings of the Thirteenth National Conference on Artificial Intelligence and Cognitive Science*. Springer, 2002. Limerick, Ireland.
6. L. McGinty and B. Smyth. The Power of Suggestion. In *Submitted to the International Joint Conference on Artificial Intelligence*. Morgan-Kaufmann, 2003. Acapulco, Mexico.
7. D. McSherry. Increasing Recommendation Diversity Without Loss of Similarity. In *Proceedings of the Sixth UK Workshop on Case-Based Reasoning*, pages 23–31, 2001. Cambridge, UK.
8. D. McSherry. Minimizing dialog length in interactive case-based reasoning. In B. Nebel, editor, *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI 2001)*, pages 993–998. Morgan Kaufmann, 2001. Seattle, Washington.
9. D. McSherry. Diversity-Consciuous Retrieval. In S. Craw and A. Preece, editors, *Proceedings of the Sixth European Conference on Case-Based Reasoning (ECCBR 2002)*, pages 219–233. Springer, 2002. Aberdeen, Scotland.
10. H. Shimazu. ExpertClerk : Navigating Shoppers' Buying Process with the Combination of Asking and Proposing. In Bernhard Nebel, editor, *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001)*, pages pages 1443–1448. Morgan Kaufmann, 2001. Seattle, Washington, USA.
11. B. Smyth and P. McClave. Similarity v's Diversity. In D. Aha and I. Watson, editors, *Proceedings of the International Conference on Case-Based Reasoning*, pages 347–361. Springer, 2001.
12. B. Smyth and L. McGinty. Tweaking Critiquing. In *Submitted to the International Joint Conference on Artificial Intelligence*. Morgan-Kaufmann, 2003. Acapulco, Mexico.