

The benefits of opening recommendation to human interaction

Eoin Hurrell and Alan F. Smeaton
CLARITY: Centre for Sensor Web Technologies
School of Computing
Dublin City University
Glasnevin, Dublin 9, Ireland
eoin.hurrell@computing.dcu.ie, alan.smeaton@dcu.ie

This paper describes work in progress that uses an interactive recommendation process to construct new objects which are tailored to user preferences. The novelty in our work is moving from the recommendation of static objects like consumer goods, movies or books, towards dynamically-constructed recommendations which are built as part of the recommendation process. As a proof-of-concept we build running or jogging routes for visitors to a city, recommending routes to users according to their preferences and we present details of this system.

Interactive recommendation, multimedia, dynamically-constructed, conversational

1. INTRODUCTION

This paper describes work being done to use user interactivity in order to make recommendation systems more conversational and self-perpetuating through incorporating user feedback as part of the process. Computers have always been very good at doing the simple repetitive tasks that people would rather not. Information retrieval is a classic example of this, with the user requiring content relevant to their query amid a large pool of possible documents. The computer is then responsible for filtering the content quickly. This is a straight-forward search task when the user is aware, even vaguely, of what they are looking for. It is recommendation when the user has no specific target in mind and just wants advice on what they might like.

2. TRADITIONAL RECOMMENDATION

Recommendation is a basic human activity that has existed in every form from “don’t go near the red berries they don’t taste good” to “I tell most people to buy the bigger version”. It occurs frequently in conversation among friends and as advice from people as diverse as co-workers to sales assistants, and is a valuable way to discover new things a person may be interested in. It is only natural that there is a place for this sort of advice in systems which have large collections of unique items that a user may be interested in. As will be shown later,

such systems can be built to not only recommend but also to give users the tools to create new content which itself can then be recommended by framing the experience as a conversation.

In the same way search systems came from the need to search through libraries of information, recommendation systems initially developed from the need to show customers products they were more likely to purchase during their interactions with online shopping systems. Recommendation systems attempt to fill our knowledge gap by mimicking the friend who advises on movies, the book critic whose opinions are always spot on or the magazine that always gives the best reviews of restaurants as described in Resnick and Varian (1997). At present, recommendation is almost as ubiquitous as search through its widespread uptake by businesses on the internet and covering all kinds of services and products. These systems are commonplace as a method for highlighting to users new items such as books, movies, websites, hotels or businesses, that will most probably be of interest or of use to them.

Automated recommendation seeks to provide users with accurate and useful recommendations of atomic entities such as a complete book or a movie, a camera to purchase, a hotel to book, etc., all within a specified and narrow domain. The technology underpinning recommendation systems continues to be based on leveraging textual metadata for

representing the entities while content-based access to non-textual media such as image and video has limited use in the operation of recommendation, though non-text entities may be the objects that are recommended.

In modeling recommendation system a common trend is to approach it using the metaphor of the shop, in order to provide a recommendation akin to a shopping assistant. The approach to this has always been to watch, to collect user activity and then link it to recommendable items in some way. The two most prominent methods of finding items a given user will probably like but hasn't yet seen are collaborative and content-based filtering (Adomavicius and Tuzhilin (2005)). *Collaborative* recommendation focuses on grouping similar users based on what they like, while *content-based* filtering seeks to form connections between users and the types of content they like mostly using the items' metadata. Various attempts to mix the two have been made (Burke (2000)) but rarely is active user engagement a priority in recommendation. In other words, the process of recommendation is effectively completed from start to finish without the user being involved in any kind of true interaction, in the sense of explicit interaction rather than implicit.

In this paper we extend the conventional recommendation process in two directions. Firstly, we turn the process of recommendation into a conversational interaction between user and system. This conversation helps to refine and focus the user's real information preferences, in much the same way that much of our information seeking activity takes place as an interactive search process anyway. We engage in this conversational interaction because the second contribution of our work is to do with the unit which is recommended. Conventionally, atomic units such as books, hotels, or electronic goods are the topic of the recommendation whereas in our work we recommend a route for a runner or jogger, that route being an aggregation of parts of other routes which in turn have their own recommendations. We thus build up the object which is recommended, the route, out of fragments of other routes combined together into a new entity. We demonstrate this with a system we have built and we illustrate its usefulness and feedback from users through a qualitative evaluation.

In the following sections our work on Run Recommender, an application of interactive recommendation which can be used to create new recommendable items based on the user's interests, will be used as an example interactive recommendation. It will show the benefits to user modeling and feedback (including new item creation) of reframing recommendation as an exploratory process in which the

user is prompted by the system to make changes. The system provides a simple hybrid recommendation based on the distance a user wants to run, their starting point, the most popular routes in the area and any landmarks the user may wish to include on the route itinerary. It provides the user with the most popular aggregate route of roughly their desired distance and then allows them to alter it. In order to calculate this, a route is broken down into a set of correspondence points and each point is assigned a popularity, that is the number of routes it occurs in. A route is therefore ranked on the sum of the popularity of its points, which has the advantage of preferring routes close to the user's desired distance. Routes are constructed from the shortest distance between each of the points on the route, in order to respect unique areas of interest on the route while preventing problems occurring in respect to rendering.

3. MOTIVATIONS FOR INTERACTIVITY

Recommendation systems almost seem to go out of their way to avoid direct interactivity ! Systems are designed to enable the user to shop without being hassled by the shop assistant, and this leads to a unique set of problems. One of the biggest is the *new user* or cold start problem. Since the system makes all of its recommendations by utilizing a user's previous actions as indications of interest, a new user is by default interested in nothing. The recommender becomes more accurate the more user information it has, as the quality of the information it retrieves is subjective and requires user details, unlike search which can be gauged by its precision and recall objectively (a problem explored by Herlocker et al. (2004)).

Information gathered from a user in order to offer better recommendations is done in different guises, items purchased or viewed can contribute, but the standard measurement used is user ratings. These ratings are typically meant to be seen by others, as an expression of user opinion on the item, and are not created with the explicit intention of improving the user's own recommendations. Even if the user is aware that the ratings they give contribute not just to the community of users but to the system itself, the connection is vague. A given rating will produce unknown changes, so the aware user experiences recommendation much like early cultures experienced magic, as an unknowable effect from a known cause.

From a user's perspective the items they are recommended are supplementary to the shopping activity they are engaged in, so this lack of direct interactivity makes sense. There are however many other situations that would benefit from a

more conversational approach, such as the type of interaction that a person would have with their friend when discussing what new movie they'll go to see. In these situations the user becomes more involved in actually finding the item they are most interested in and potentially provides much more information about themselves and their preferences to the recommender. This feedback is invaluable in making the system able to provide more accurate item filtering quickly, and engages the user. If the conversational model is followed it will provide new information about the options to the user, giving them the opportunity to learn and make better decisions about items, and the user can contribute meaningfully to the system.

4. THE RUN RECOMMENDER

The system built to test our approach to conversational recommendation is an exercise builder, designed to provide runners with a tailored running route in an unfamiliar city or other location. A web application using Google Maps was built in order to present the route based on aggregation from one or more other routes, and also includes other supporting multimedia information. A collection of pre-loaded running routes was scraped from a popular running route website, each route being rated based on its distance and the overlap of its individual GPS points among other routes, which allowed us to create a measure of popularity for each route based on this overlap. The system begins with a default route recommendation displayed to the user, which is then modified by the user as needed, based on conversational interaction.

The exercise builder uses a simple collaborative recommendation engine that asks the user to confirm their current location as their starting point and to specify a distance they are prepared to run. It then refreshes the displayed route to reflect a new recommendation from the same default starting point. The user may alter the starting point or target running distance to have a new route recommended. This new route can be modified using seven equidistant drag-points along the given path. This allows for minor changes as well as significant ones.

Publicly-available geo-tagged images from the photo management website Panoramio¹ are placed on the map using their location data to show the area of the suggested route in more detail, and provide visual information on possible areas or landmarks of interest such as monuments, buildings or other interest points that the user may want to see. By displaying the 50 most popular (as rated by

Panoramio users) images in the visible area the user gets a representative sample of the sights to be seen. This functions to both elicit interest and allow users to interact with their recommendation in a meaningful way by providing a context within which they may want to vary the generated route.

4.1. The Hybrid Recommendation System

Our recommendation system is based on the 1,302 most popular running routes in the Dublin, Ireland, area scraped from a popular run-tracking website. The initial recommendation to a user as shown in Figure 1 is the nearest and most popular route available that is not any longer than the distance the user has specified to run. Routes are plotted as the shortest path between 8 equidistant drag-points from the available run route, 8 being a technical limitation of Google Maps Direction service, which works out the best route between points. Using this method allows the user to alter the route by dragging any of the drag-points to another place, rerouting the path to that place.

No metadata or map annotation currently exists for the different types of running surface on a route (grass, woodland, tarmac, concrete, etc.) and this is sometimes important to runners, so the calculated value for popularity is used in an attempt to recommend routes that do not travel along unsuitable paths (motorways, hazardous pedestrian areas etc.). The recommendation is built using the user's target running distance and the starting point provided. It is assumed that the user starts and finishes at the same location, i.e. it is not a point-to-point route s/he is looking for. Using this information the system generates a path by taking points from the most popular known route that is not longer than the user's stated preferred distance.

4.1.1. User Information Requirements

The exercise route-builder system models users after its database of cases, and provides recommendations using minimal pre-gathered information. As stated earlier, a default recommendation initially appears, allowing the user to alter the recommendation without even filling in their desired distance, moving the focus to interactivity over static recommendation.

4.1.2. Route Information Requirements

Each point in every route has a popularity score that is the measure of its re-use across every route. A route's popularity is the total popularity of each of its points, which remains unnormalized to maximize both distance and popularity in the recommendation. This allows recommended routes to be close to the specified distance given a relatively small number of routes over an area. Along with this the total distance of each route as well as its elevation profile, is known.

¹<http://www.panoramio.com>

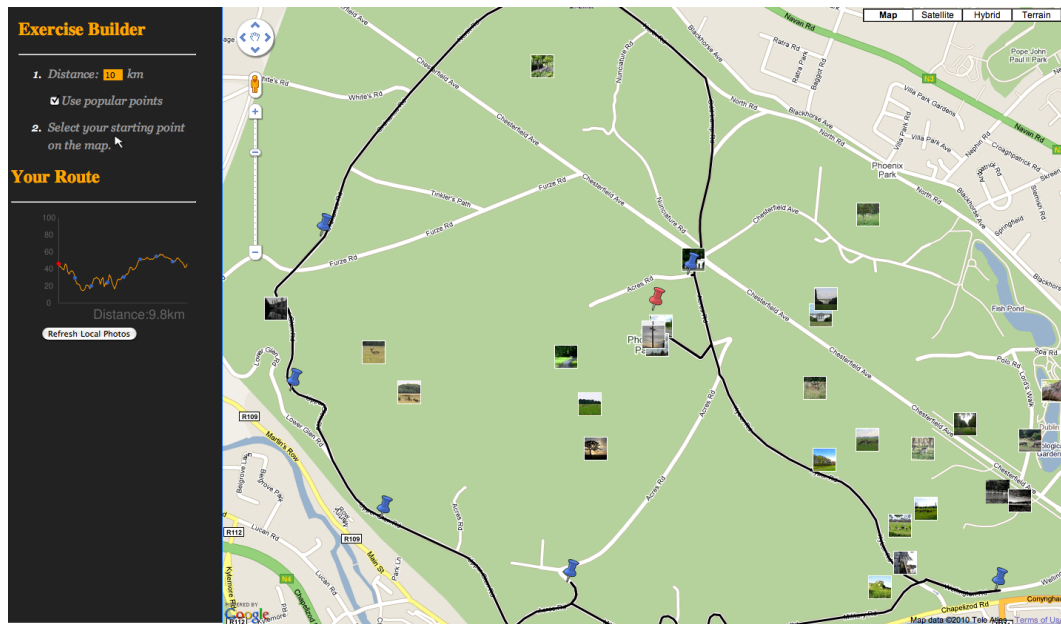


Figure 1: A recommended route including drag-points and multimedia content.

4.2. Multimedia Exploration

Allowing the user to explore the areas surrounding their route recommendation is important, but without multimedia overlays there is a lack of any of the detail that users would look for to provide reasons to alter their recommended route. Using a layer of photo thumbnails that can be enlarged and browsed allows the user to see various aspects of the area that has been recommended for them. A publicly available service embeds images at the geographic coordinates at which those images were taken. Pictures are shown, where available, for every stage of the route recommended as well as the surrounding alternatives as shown in Figure 2. This provides incentive to either reaffirm their choice, or alter it. This interactivity attempts to mimic the inquisitive exploration a person might have when discussing an area unknown to them with someone in a position to recommend places to visit or to run through. It is also useful to provide visual cues of the area to those unfamiliar with it.

5. METHODS OF INTERACTIVITY

Interactive recommendation has been previously explored to create systems (such as the one outlined in Alon et al. (2009)) that engage the user as fully as possible. Shimazu (2001) outlined a system designed to create the experience of consulting a shop clerk who suggests items based on need, while the Adaptive Place Advisor (detailed in Langley et al. (2004)) showed the power of conversational models in developing user models quickly to provide satisfactory items. Both systems were designed

to elicit user sentiment toward attributes of items, adding depth to the recommendation through direct interaction with the user. In making interactive systems there needs to be a clear understanding that the user can effectively be *mined* for more data by just asking.

5.1. Conversational Style

In the Run Recommender a simple interactive approach was taken. While most recommendation algorithms are closed to an interactive flow the Run Recommender extends the process by allowing the user to divert the path and examine the effects it will have in order to find a satisfactory route. The interactive recommendation process is iterative, with the user being offered a recommendation and data related to the closest alternatives that are not considered optimal. From this the user can alter the recommended route into something new, comprised of points recommended to them and points they define, connected via shortest-path routes.

The architecture for our route recommendation system depends on engaging the user in an interactive process, which represents a shift from the usual application of such recommendation being a feature added to a larger system. In contrast to other systems such as McGinty and Smyth (2003) or Göker and Thompson (2000), our system establishes a conversational style by having a linear ask-respond style conversation, thus reducing the recommendation space. The result is that in a system such as the one outlined below, the user can effectively create new items (routes) that would not otherwise be recommended. It also

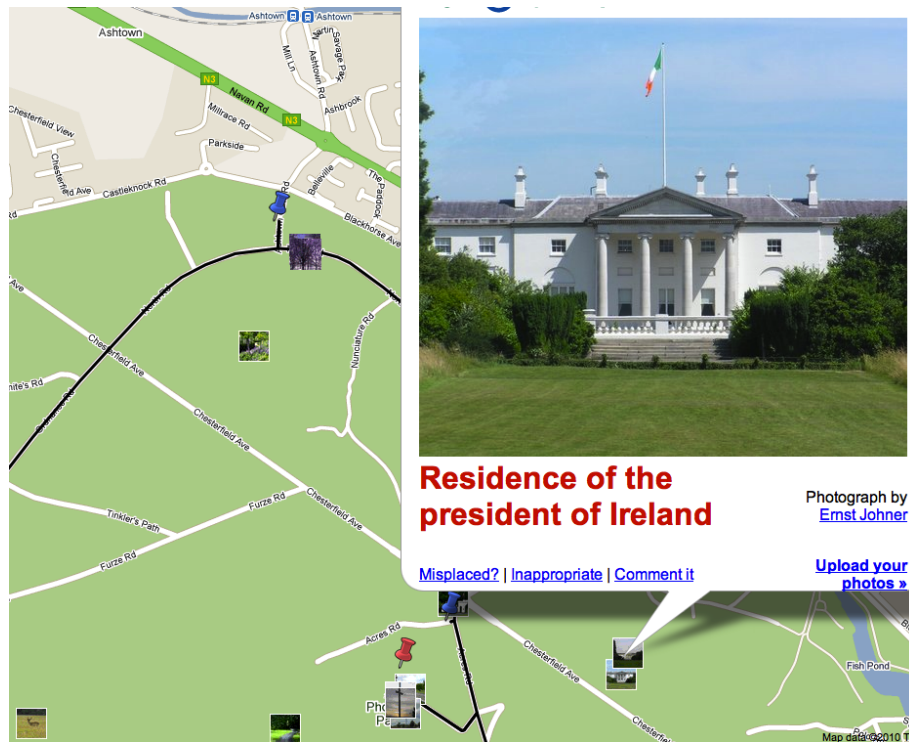


Figure 2: Multimedia content exploration in action.

seeks to allow the user to guide the process more fully using multimedia elements. In this way the user benefits from an increased knowledge of the recommendation space and is thus more fully informed as to the quality of their recommendation. This addresses one of the drawbacks of conventional recommender system applications, the issue of how to resolve question in the users' mind of *why* something is being recommended. Sometimes, feedback along the lines of "Users who bought X also bought Y and Z", just isn't enough.

By making recommendation the focus of the system the user is actively engaged in finding the best possible route for them, allowing them to establish how they are different from other users. This has the dual advantage of improving user satisfaction with recommendations and also providing more information to quickly refine user groupings for new and existing users.

Since the architecture is designed to focus on post-recommendation refinement and information solicitation, the pre-recommendation information requirements can be relatively simple, indeed the system can benefit from a certain 'pacing' of information gathering, with too much initial form-filling becoming tedious and hindering usage. The ideal format mimics a conversation, with the user providing the system with a relevant piece of information such as, 'I do prefer running on grass so Central Park (New York) would be good to include' or

'I've already seen the Colosseum last time I was in Rome' and the system renewing its recommendation to reflect this.

At the end of the interaction process, the user has worked to create is a new route. This new route is potentially different from those available in the current itemset in a number of ways. It could represent an already existing route that was considered less than optimal by the system, in which case storing the route will increase the popularity of its component points and the prospective route will be more likely to be chosen in future. It could also betray the user's initially stated requirements, by being a greater distance than the stated desire. Most interestingly the new route could be unique and not exist in any form in the set of known routes, in which case the user has helped to create a valid item that is entirely new to the system, allowing the itemset to be extended to offer new routes to the user. In every case the interaction benefits the system by providing a large amount of information not traditionally collected that can be put to good use to benefit the entire userbase. In this way the system is designed to be self-perpetuating and to quickly find good recommendations for even new users.

The workings of recommendation systems have always depended greatly on their unique problem space, making a 'silver-bullet' recommendation system, let alone one that is interactive, unlikely. System design is usually affected by the nature

of the items, the ratio of users to items and whether they use collaborative, content-based or hybrid techniques. Different itemsets benefit from different approaches in order to produce the better recommendations, due to factors such as strength of opinion on the topic, which makes comparing the relative ability of different systems difficult. In this field interactivity must be integrated carefully in order to make appropriate additions to the process.

6. WORK TO BE DONE

Our plans for interactive recommendation are to explore the means by which it can be used in a number of contexts to expand on and improve recommendation through user engagement. Following our initial work into capturing and using direct feedback from the user, more work is now needed to explore the various contexts in which this will work. It is also hoped that the conversational recommendation model can be extended to give the user a more concrete impression of agency behind the system's responses and queries, such as that found in common-sense reasoning systems. The hope that a homogenous approach to interactive recommendation can be found through user contribution that will make it easier to benchmark the success of recommender systems objectively.

Overarching all this is the requirement to complete some form of evaluation or benchmarking. Measuring the effectiveness of our techniques cannot be accomplished using the usual recommender system benchmarks since the whole premise of our work is that it is interactive, and that cannot be replicated except in scenarios with real users. We have already completed a qualitative user study, measuring feedback from users based on their subjective judgements the results of which are presented elsewhere. Users felt the runs provided were apt and the inclusion of multimedia interaction was an improvement. The system described in this paper, the Run Recommender, is shown to encapsulate all of the elements of an interactive recommendation system and has helped to crystallize our own thoughts on how this model of user-system interaction should proceed.

ACKNOWLEDGEMENT

This work is supported by Science Foundation Ireland under grant 07/CE/I1147.

7. REFERENCES

Adomavicius, G. and Tuzhilin, A. (2005) Toward the next generation of recommender systems: A survey

of the state-of-the-art and possible extensions, *IEEE Transactions on Knowledge and Data Engineering*, 17(6).

Alon, N., Awerbuch, B., Azar, Y. and Patt-Shamir, B. (2009) Tell me who i am: An interactive recommendation system, *Theory of Computing Systems*, 45, 261–279.

Burke, R. (2000) Knowledge-based recommender systems. *Encyclopedia of Library and Information Systems*, Marcel Dekker.

Langley, P., Thompson, C.A. and Göker, M.H. (2004) A personalized system for conversational recommendations. *Journal of Artificial Intelligence Research*, 21, 393–428.

Göker, M. and Thompson, C. (2000) Personalized conversational case-based recommendation. *Advances in Case-Based Reasoning*, Springer LNCS, 1898, 29–82.

Herlocker, J.L., Konstan, J.A., Terveen, L.G. and Riedl, J.T. (2004) Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 5–53.

McGinty, L. and Smyth, B. (2003) On the role of diversity in conversational recommender systems. *Case-Based Reasoning Research and Development*, Springer LNCS 2689, 1065–1065.

Resnick, P. and Varian, H.R. (1997) Recommender systems. *Communications of the ACM*, 40(3), 56–58.

Shimazu, H. (2001) Expertclerk: navigating shoppers' buying process with the combination of asking and proposing. *Proceedings of the 17th international Joint Conference on Artificial Intelligence*, Morgan Kaufmann, Volume 2, 1443–1448.