Recommending Customized Trips Based on the Combination of Travel Regions

Wolfgang Wörndl, Michael Ludwig, and Daniel Herzog

Department of Computer Science
Technische Universität München, Germany
{woerndl,ludwigm,herzogd}@in.tum.de

Abstract

In this paper, we present an approach to recommend travel regions for independent travellers. We have developed an algorithm based on an approximation for the knapsack problem with extensions to be able to combine multiple regions to a composite trip. We have also developed an innovative web platform to present results to the user. The web platform allows for easily swiping through the recommendations, which are represented in one full screen each and automatically integrates pictures from Foursquare. Both parts of the project (recommendation algorithm and web platform) were successfully tested in small scale user studies.

Keywords: web platform, recommender system, tourist trip design problem, travel, knapsack problem

1 Motivation

Consider the following scenario: You want to travel in September for four weeks with a budget of 2000€, you like nature and hiking with some cultural highlights as a plus, where should you go? While similar services exist for choosing among travel packages for organized trips, little is available to support independent travellers when selecting their destinations.

The idea of this project is to provide a platform where users can specify their travel requirements and preferences (time of year, duration of trip, budget etc.) and get a list of recommended options, i.e. travel regions that suit their query best. One of main requirements is that the solution should be capable to combine two or more partial trips or regions to a longer itinerary in a reasonable way, e.g. combining nearby regions.

In this paper, we first outline related work and briefly introduce our data model. Then we explain the recommendation algorithm in Section 3. In Section 4, we present our web platform before concluding the paper with a brief summary and outlook.

2 Background

Existing scientific literature covers topics such case-based and other recommender systems, (package) travel planning, modern user interfaces and scalable, flexible software architecture for web applications, but there is little work on combining these topics to support independent travel planning. Ricci et al. (2006) present an approach called Trip@dvice which integrates case-based reasoning, interactive query management and collaborative filtering. They developed two web-based prototypes called NutKing and DieToRecs.

Liu et al. (2011) addresses the issue of sparse user rating data for travel packages and how to generate personal recommendations out of past data. It focuses on already existing travel packages from travel agencies. Xie, Lakshmanan and Wood (2010) cover the topic of composite recommendations and propose a mathematical algorithm to solve this task. They focus on the recommendation on top-k packages with approximation algorithms because the exact problem solving is NP-complete.

For our project, we developed a hierarchical data model consisting of travel regions. In our model, a region is always a sub-region of another region while the world is the parent region of every region. Regions contain the necessary information for recommendation: recommended months for travel on a 5-point Likert scale, minimum and a maximum recommended duration of typical trips and budget information for regions, crime level and other information (Herzog and Wörndl, 2014). We have also modelled traveller types such as *Free Spirits* or *Cultural Explorer*, which were inspired by a market segmentation tool of the Canadian Tourism Commission (http://en-corporate.canada.travel/resources-industry/explorer-quotient). The region attributes include how good the region matches a traveller type. Our dataset consisted of 152 travel regions; the data was manually compiled from various Internet and other sources.

3 Recommendation Algorithm

Composite trip recommendation can be seen as a special case of the knapsack problem to find best combinations of multiple travel items (Xie, Lakshmanan and Wood, 2010). The underlying idea is to combine as many single travel items like regions, routes or activities as necessary to maximize the benefit for the user while still respecting existing limitations like time and money. In our case, the problem gets more complicated than the standard knapsack problem because the value of a region is not only determined by the user query. Rather, it depends on the presence or absence of other regions in the recommended composite trip. This extension of the knapsack problem is called the Oregon Trail Knapsack Problem (Burg, 1999). Imagine a travel sequence that recommends visiting Germany, Austria and Switzerland. If the user could spend more time and money, the system can add further regions to this trip. Depending on the user's preferences, additional regions in or close to Central Europe should be recommended. Only exceptional circumstances legitimate an additional region far from Central Europe because the effort of visiting this region during this trip is disproportional.

We developed an algorithm based on an approximation for the knapsack problem and extended it to recognize dependencies like the distance between the regions while calculating best combinations. To ensure a high diversity, the algorithm is able to calculate the optimal duration of stay per region in the composite trip. The algorithm is composed of three phases:

1. Reduce number of regions

2. Rate regions

3. Calculate the best combination of regions

The first phase takes advantage of the hierarchical structure of our travel database. If the user wants to exclude one or more regions, all sub-regions are removed from consideration as well. For instance, if the user wants to travel through Europe but already explored Scandinavia, there is no need to execute any calculations for other continents or any Scandinavian regions.

During the second phase, the algorithm rates every region that can be considered for a recommendation in order to determine the value for the knapsack problem approach. This value determines how well a certain region fits the user's requirements when regarding it exclusively and without other regions in a possible composite trip. As the description of regions in our database is composed of a well-defined set of features (cf. Section 2) we can use case-based similarity metric to assess how well a region fits the user query (Smyth, 2007). Before executing the third and last phase, we exclude regions with a value below a threshold in order to prevent composite trips including regions that do not satisfy the user's demands but could be cheap enough to be considered in a recommendation only for the use of free capacities.

In the end, the values are used as input for the approximation for the knapsack problem. We implemented a dynamic programming solution that adapts the region values and thus the value of the composite trip depending on the presence or absence of other regions in the trip. In our case, the value of an item is determined by the case-base recommender (step 2). We split every week in one-week-blocks before executing the algorithm and decrease the value of every following week in order to determine the optimal duration of stay per region and to ensure diversity. The dynamic programming approach calculates an approximation by iterating over the number of available regions, as well as over the limits. We use two different limits: budget in Euro and maximum duration in weeks.

We have tested the approach by having an expert rate the outcome of the proposed algorithm and two simpler baseline algorithms. The baseline approaches did not take our extensions to the knapsack problem into account, e.g. weekly penalties. Our travel recommendation algorithm resulted in the highest overall satisfaction to the expert. Furthermore, it is rated best in the metrics *regions fit together* and *routing*. In terms of diversity, our travel recommendation algorithm is rated somewhat lower than the two comparative algorithms. Further effort is necessary in order to understand the preferences of potential users and to find out how a better diversity can be ensured without recommending regions that fit together less. One possibility is extending and refining the penalty function in our algorithm. Detailed information about the recommender algorithm and its evaluation is available in (Herzog and Wörndl, 2014).

4 Web Platform

To present the recommended travel regions to the user, we have developed an innovative web platform. The focus of the web application was to provide a high usability and the ability to present a travel package easily to the user after creating a small traveller profile and expressing a basic query. The display of recommended travel regions includes user-generated pictures from Foursquare. Users can easily show their affection by saving a recommendation for later in-depth review ("Merken" in Fig. 1) or aversion by clicking the dislike button ("Mag ich nicht" in Fig. 1). The user feedback could also be used to improve the recommendation algorithm but this is not implemented yet.

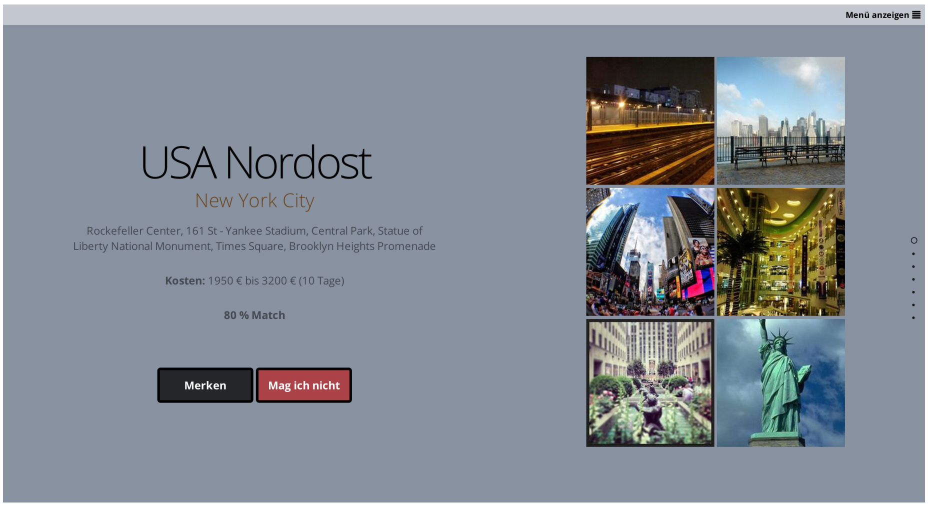

Fig. 1. Screenshot of the *onepagescroll* recommendation view

On the first use, the user has the ability to create a small personal profile and define some attributes in a wizard-like menu. One of the main ideas was the identification of a traveller type with a travel personality which helps in finding suitable places. After a user profile was created, the user can easily express search queries by defining how many days he/she wants to travel, how much budget he/she has available and in which month to travel. After the recommendation algorithm is run, the application shows the recommended travel regions. On the result page, the user can easily swipe through all the recommendations. Each recommendation is represented in one full screen to allow the user to get a good impression of the region with illustrating pictures (Fig. 1). We call this *onepagescroll* view. He/she can rate them or just skip them by using a *two-finger-swipe* gesture. This interface is heavily inspired by UX patterns from mobile devices. After rating and saving some recommendations, the user has the possibility to go to a final review step where he/she can see all recommendations in detail on one page.

The platform additionally offers a user interface for travel experts to enter and modify data about countries, regions, routes and sights. The tool also allows for entering connection information between different regions. Image data does not need to be added as it gets dynamically matched through the Foursquare API. Foursquare already contains a lot of venues around the world and highly ranked pictures get picked automatically. The application was implemented with a modern state-of-the art software architecture. It uses the full stack Play framework and Scala as programming language. Data is stored in the NoSQL database MongoDB. This combination of technologies allowed for a rapid application development in an agile manner. The

Play framework in combination with MongoDB allows for a high scalability. MongoDB with its schema-less storage allows for a fast development optimized for performance.

The user interface was evaluated in a small qualitative study that proved the usefulness and advantages of showing pictures besides the recommendations. The users could identify themselves more with the travel regions and the offered sights and were more likely to rate them positive. In general most of the users were happy with the workflow of the application and missed such a tool in their daily lives.

5 Conclusion

In this paper, we have outlined a travel recommendation algorithm that addresses the Oregon Trail Knapsack Problem and applies it to the travel domain. The approach is able to combine multiple travel regions for composite trips. To present the information to users, we have developed a web platform that allows for easy specification of input data and appealing presentation of results to potential end users. The platform integrates image data from external sources. We have conducted two separate small scale studies that, first, showed the usefulness and applicability of the proposed recommendation algorithm and, second, proved that the used UI components enhanced the usability of the web application.

Future work includes extending the system to recommend routes or specific itineraries in addition to travel regions. We also need to improve the performance of the presented recommender algorithm since some queries take too much time for an online web application. Therefore, we used a simpler version of the algorithm for the test of the user interface. We then plan to conduct a more extensive user study with potential users in order to test the whole system.

References

Burg, J.J., Ainsworth, J., Casto, B. & Lang, S.-D. (1999). Experiments with the "Oregon Trail Knapsack Problem". *Electronic Notes in Discrete Mathematics*, 1:26-35.

Herzog, D. & Wörndl, W. (2014). A Travel Recommender System for Combining Multiple Travel Regions to a Composite Trip. *Proc. New Trends in Content-based Recommender Systems (CBRecSys) workshop*, *8th ACM Conference on Recommender Systems (RecSys 2014)*, Foster City, CA, USA.

Liu, Q., Ge, Y., Li, Z., Chen, E. & Xiong, H. (2011). Personalized travel package recommendation. *Proc. 2011 IEEE 11th International Conference on Data Mining*, 407–416.

Ricci, F., Cavada,, D., Mirzadeh, N., Venturini, A. (2006). Case-based travel recommendations. Fesenmaier, D.R., Werthner, H. & Wöber, K. (eds.) *Travel Destination Recommendation Systems: Behavioral Foundations and Applications*. CAB Publishing.

Smyth, B. (2007). Case-based recommendation. Brusilovsky, P., Kobsa, A. & Nejdl, W. (eds.): *The Adaptive Web*, Berlin-Heidelberg: Springer, 342-376.

Xie, M., Lakshmanan, L.V. & Wood, P.T. (2010). Breaking out of the box of recommendations: from items to packages. *Proc. of the 4th ACM Conference on Recommender Systems (RecSys 2010)*, ACM Press: 151-158.