Exploiting Web Analytics Tracking for Bootstrapping a Case-based Recommender System

Paolo Massa[a], Adriano Venturini[b,] and Michela Ferron[a]

[a] Fondazione Bruno Kessler, Italy
{massa,ferron}@fbk.eu

[b] eCTRL Solutions, Italy
venturini@ectrlsolutions.com

Abstract

Case-based recommender systems have been successfully applied to tourism web sites for suggesting to travellers products they might like such as hotels or events. Since they exploit previous experiences by other travellers (cases), their casebase needs to be bootstrapped at deploy time by inserting initial experiences. In this paper we address this open problem and propose a methodology for bootstrapping case-based recommender systems by exploiting the tracking features of Web Analytics tools.

Keywords: bootstrap; recommender system; case based reasoning; web analytics, custom variables.

1 Introduction

In the domain of tourism, an important open problem is related to travel information search and retrieval. Case-based reasoning (CBR) is a technique used for creating recommender systems able to suggest tourism products such as hotels or events by searching and retrieving how other people previously fulfilled their tourist needs.

However CBR recommender systems need to be bootstrapped before they can work properly and effectively: since they rely on past experiences by other travellers, this previous knowledge must be created when the system is firstly deployed. This is particularly challenging in the domain of tourism in which instantiating the recommender system for a specific location in the world could rely on very different information when deployed for a different location in a different context, with different tourist patterns, attractions and locations.

Often CBR recommender systems are deployed as an integrated part of tourist web sites dedicated to a specific location, hotel or event. In this paper we propose a solution for bootstrapping a case-based recommender system that exploits the tracking of web sessions already offered by a Web Analytics tool. In particular, we describe the solution we implemented on the web site of the travel agency of Trento, north Italy, that is accessible at discovertrento.it.

2 The open problem: bootstrapping of a casebase

A case-based reasoning (CBR) recommender is a knowledge-based system that builds a recommendation basically exploiting a “search and reuse” approach. CBR is able to utilize the specific knowledge of previously experienced, concrete problem situations (cases). A new problem is solved by finding a similar past case, and reusing it in the new problem situation (Aamodt & Plaza, 1994). CBR recommender systems have been used in many different domains (Aamodt & Plaza, 1994) and they have been proved particularly successful in the domain of tourism (Ricci et al., 2006).

In this paper we build on our previous work, the case-based recommender system Trip@dvice (Ricci et al., 2006). Trip@dvice supports the selection of travel products (e.g., a hotel or a visit to a museum or a climbing school), and building a travel plan, that is a coherent (from the user point of view) bundling of products. The recommender system built into Trip@dvice produces ordered lists of travel products as suggestions. In this paper we present a component that was missing in our CBR recommender system: a methodology for bootstrapping the case-base. This is important because the quality of the recommendations generated by a CBR recommender system depends heavily on the quality of the cases stored in its casebase. Intuitively, if there are no good travel plan (cases) in the casebase, the selection and ranking algorithms cannot work well because they cannot select any good travel plan.

3 The proposal: a method for bootstrapping CBR recommender systems exploiting web analytics tracking

In Trip@dvice, the casebase is composed of travel plans. A case includes both components that represent the search/decision problem definition, i.e. the travel's and travellers' characteristics, and the problem solution, that is the set of products included in the plan.

More precisely, a case comprises the following two components: collaborative features and products cart. Collaborative Features (CF) are features that describe general user's characteristics, wishes, constraints or goals (e.g. desire to relax or to practice sports). They capture preferences relevant to the user's decision-making process, which cannot generally be mapped into the features of products in the electronic catalogues. These features are used to measure case similarity. The Products Cart (PC) contains the set of products chosen by the user during the recommendation session represented by the case. The products comprise some transportation services (flight, train or car), accommodations, monuments. Products are associated with characteristics such as “suited for family” or quantity of stars for hotels. Travel plans are stored as cases in the repository (casebase).

We briefly describe the entire system for bootstrapping the Trip@advice casebase and then focus on the part that exploits Web Analytics tracking during the first step of the system. The system is composed of two steps (see Figure 1). The first step produces a set of cases in which only the part of collaborative features is filled and they roughly represent tourists with their needs for products. The second step consists in filling the cases with products that can satisfy these needs and is done using rules entered in the system by the casebase admin.

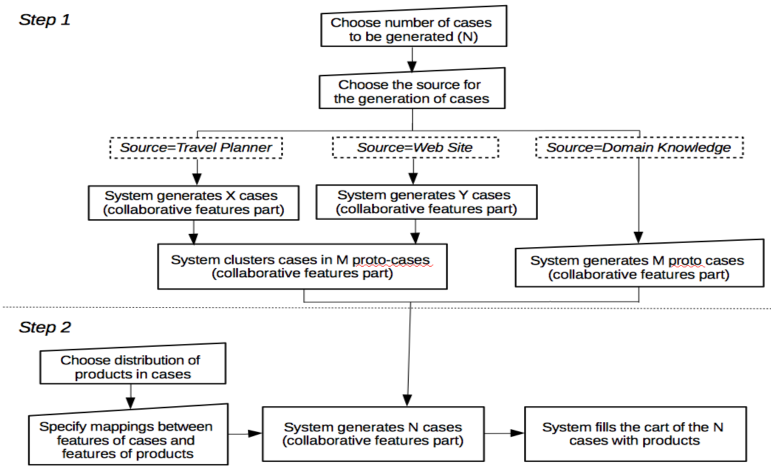

Step 1
Choose number of cases to be generated (N)
Choose the source for the generation of cases
Source=Travel Planner
Source=Web Site
Source=Domain Knowledge
System generates X cases (collaborative features part)
System generates Y cases (collaborative features part)
System clusters cases in M proto-cases (collaborative features part)
System generates M proto cases (collaborative features part)
Step 2
Choose distribution of products in cases
Specify mappings between features of cases and features of products
System generates N cases (collaborative features part)
System fills the cart of the N cases with products

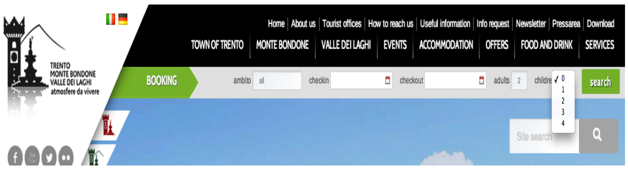
Figure 1: the bootstrapping system for the CBR recommender system Trip@dvice.

During step 1, cases can be generated using information coming from three different sources: (1) the travel planner itself, with the information about travel and traveller such as composition of the travel party or country explicitly entered by the users themselves through the interface of the travel planner, (2) a web analytics tool deployed on the web site, tracking the different web sessions, and (3) the manual coding of tourism experts via the tool interface, able to create the collaborative features part of cases representing prototypical tourists of the specific destination. Options 1 and 2 tend to generate a large number of cases and hence requires a clustering step in order to extract few prototypical tourists with their relative importance weight that is the input of the second step. The clustering phase (not described here) is similar to what (Pitman et al., 2010) proposed; this approach falls under the idea of using clustering for creating clusters of tourists (Dolnicar, 2008).

In this paper we describe option 2, the use of web analytics tracking. This is particularly interesting since it does not require asking users to use the travel planner for entering information but without receiving recommendations (as in option 1 when bootstrapping the recommender) and because the web analytics tools are usually already in place on the tourist destination web site. With option 2, it is possible to deploy the web site without travel planner, collect web sessions for a certain period of time (for example, two months) and then use the collected web sessions for bootstrapping the casebase and hence being able to deploy the CBR recommender system, now completely working and producing recommendations, on the web site.

Web analytics tools are used for tracking visits to a web site in order to then analyse how visitors access the web site (Kaushik, 2009). The most used web analytics tool is Google Analytics, which is software-as-a-service and hence does not need to run on your local server but, by simply adding some javascript on your web site, stores web sessions on Google servers and provides an interface with statistics about sessions. Typically web analytics tools stores (and provide graphic reports about) information such as which nation and city visitors come from, which operating system, device and browser they user, which pages they visit (and hence what are the most visited pages), which links from other web sites visitors follow for arriving at your web site, and more. The different information tracked by web analytics tools are called variables (Kaushik, 2009). Piwik is an open source software which offers functionalities very similar to Google Analytics but that can be installed on your server so that you fully control the recorded data (Piwik, 2014). Typically with web analytics tools it is possible to track and store also additional variables (custom variables) by writing ad-hoc javascript code that extracts the information from an interaction with the web site and store it with the specific session, or single visit of a page, or entire navigation history of a user: this is the scope of the custom variable (Kaushik, 2009).

In the following we describe how the tracking works with an example taken from discovertrento.it, in which we deployed the system. On discovertrento.it, visitors can search for accommodations and events specifying the local area, the checkin and checkout dates, the number of adults and children in the party (see Figure 2). TravelParty is one of the collaborative features of a travel plan and can take values such as single, family, friends, couple or group. In order to exploit web analytics, we wrote ad-hoc javascript code that is executed when the visitor clicks on the “search” button and takes the values of the different fields in the form. Conditions such as “adults=2 and children>=1” are used to assign “family” as value of a custom variable named TravelParty, just as the corresponding collaborative feature in the case, and store it together with the information about the specific web session, the scope of custom variables is hence at session level. The idea is that every web session will constitute a travel plan (just for the collaborative features part). As another example, a value of TravelParty “single” is assigned when adults equals 1 and children equals 0.


TRENTO
MONTE BONDONE
VALLE DEI LAGHI
atmosfere da vivere
Home | About us | Tourist offices | How to reach us | Useful information | Info request | Newsletter | Pressarea | Download
TOWN OF TRENTO | MONTE BONDONE | VALLE DEI LAGHI | EVENTS | ACCOMMODATION | OFFERS | FOOD AND DRINK | SERVICES
BOOKING
search
Site search

ambito: all
checkin:
checkout:
adults: 2
childre: ✓ 0 1 2 3 4

Figure 2: Screenshot of discovertrento.it. The form on top is the one from which the values of custom variables are extracted.

In a similar way, the collaborative feature "season" and "month of departure" are extracted from the custom variable "checkin", that is the date for which the travel information are searched. The feature "duration of holidays" is computed as difference between the dates of checkout and checkin. The feature "country" can be taken from the default variable of web analytics tools which compute it based on the IP address of the computer requesting the web pages. Extracting others collaborative features, such as budget and tourist interests, is more complicated and is done by saving all the pages visited by the visitor of the web site during a session and then, when the session ends, by computing approximate budget based on the costs of the visited accommodation pages and the meta-tag indicating for whom accommodations and events are suited for. For example, if a visitor visits mainly 5 stars hotels, a value of "high" is associated with her "budget" custom variable, and if she visits mainly events and accommodation whose meta-tag is "relax", relax can be added to the interests custom variable. Age and gender are two additional collaborative features and they can be inferred enabling the demographics and interests reports of Google Analytics or by explicitly asking the visitor.

4 References

Dolnicar, S. (2008). Market Segmentation in Tourism. In A. Woodside & D. Martin (Eds.), Tourism Management, Analysis, Behaviour and Strategy. Cambridge: CABI.

Ricci, F., Cavada, D., Mirzadeh, N., Venturini, A. (2006). Case-Based Travel Recommendations. In D. R. Fesenmaier, H. Werthner and K. Wöber (editors), Travel Destination Recommendation Systems: Behavioral Foundations and Applications.

Pitman, A., Zanker, M., Fuchs, M. & Lexhagen, M. (2010). Web Usage Mining in Tourism – A Query Term Analysis and Clustering Approach.

Piwik. (2014). Retrieved September 2, 2014, from http://www.piwik.org

Piwik Reporting API. (2014). Retrieved September 2, 2014, from http://developer.piwik.org/api-reference/reporting-api

Kaushik, A. (2009) Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity. Wiley.

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. Artificial Intelligence Communications, 7(1), 39-59.

5 Acknowledgements

This work has been partially funded by Travel Monitor Project, FESR 2012 funding programme, CUP C67I12000030008.