Designing Recommender Systems for Tourism

Tobias Berka
Manuela Plößnig

Salzburg Research
Jakob-Haringer Str. 5/III
5020 Salzburg
Austria
{tberka, mploessnig}@salzburgresearch.at

Abstract

In this paper, we argue that the process of developing travel recommender systems (TRS) can be simplified. By studying the application domain of tourism information systems, and examining the algorithms and architectures available for recommender systems today, we discuss the dependencies and present a methodology for developing TRS, which can be applied at very early stages of TRS development. The methodology aims to be insightful without overburdening the project team with the mathematical basis and technical detail of the state of the art in recommender systems and give guidance on design choices to the project team.

Keywords: Recommender Systems; Travel Recommender Systems; Artificial Intelligence; Software Design.

1 Introduction

Recommender systems are the technical response to the fact that we frequently rely on other peoples' experience and recommendations when confronted with a new field of expertise, where we do not have a broad knowledge of all facts, or where such knowledge would exceed the amount of information humans can cognitively deal with. Recommendations are a common means of planning in the fields of tourism, travelling and hospitality. This observation in the real world suggests that recommender systems are an intuitive and valuable extension to tourism information systems. However, recommender systems have a very mathematical and complex background as their roots are still in the area of Artificial Intelligence (AI). This means that designers and developers of recommender systems not only face the common problems of software design, but also several specific problems associated with AI applications such as the need for a good (application specific) description of the problem space and the higher risk associated with ”one-off” applications that lack the software engineering experience embodied in more traditional systems (e.g. payroll, customer management, etc).

1.1 Problems and Challenges in Tourism

Products and services in the field of tourism (like hotel rooms, packages, etc) are mainly not physical and typically exists mostly as information. For this reason, they are predestinate for electronic sale. ICT allows easily to present tourism offerings with richer descriptions to enable travelers to make more informed choices. As consequence, the complexity of product descriptions is growing (see (Werthner & Klein, 1999)).

But the tourist also claims the benefits of the Internet. Users choose their destinations among various channels and compare tourism offerings critically. The increasing use of ICTs in tourism services allows tourists to take a more active role in the production of tourism products, being no longer satisfied with standardized products. The "postmodern tourist"' with differentiated life-styles (e.g. shorter trips), individual motives (e.g. business travelers, elderly persons, culture tourists, day-tourists) and specific interests (e.g. focus on special sports) demands products tailored accordingly to stated preferences (as in (PRISMA, 2001)).

The tourists of today are very demanding and have complex, multi-layered desires and needs. They are flexible, often experienced in travelling and demand both perfection and diversity. In consequence tourist offers should be multi-optional and of high quality (discussed in (Smeral, 2003)). Further consumer needs change rapidly and requires products with shorter life cycles (see (PRISMA, 2001)).

It is the increasing use of ICTs in tourism services that allows tourists to take a much more active role in selecting and comparing tourism products. But travel information search is a complex and dynamic process. Travel information search and planning, and supporting these decisions on the Internet, becomes more and more prevalent nowadays, calling for modern means of decision-making support.

1.2 Examples of Tourism Applications

Travel recommender systems (TRS) attempt to emulate offline travel agents by providing users with knowledgeable travel suggestions to facilitate their decision-making processes. Using recommender systems, we assume that a user's needs and constraints can be mapped into a specific set of alternatives from which the user will be able to choose, based on the deployment of appropriate algorithms.

Most travel recommender systems depend on just one of the filtering approaches - for instance Triplehop's TripMatcher and VacationCoach's Me-Print uses a content-based approach, and the research project Dietorecs opts for case-based reasoning (Ricci, 2002). However, using one filtering technique can fall short when trying to make recommendations for complex products. For a TRS, many factors including individual interests, holiday season or income, must be considered when generating meaningful recommendations. Furthermore, most of the systems appear to be confined to producing recommendations only at the destination level, without providing support at a finer scale.

Many current recommender systems struggle to capture user needs, and companies have implemented different approaches to tackle this issue. Several attempts to combine collaborative and content-based filtering in other domains and recently, also in some travel recommender systems ((Pazzani, 1999; Delgado & Davidson, 2002)), can be found. It seems to be clear that only a combination of different filtering techniques can offer significant improvements on the decision-making process.

1.3 Recommender Systems

Recommender systems are an attempt to mathematically model and technically reproduce the process of recommendations in the real world (see (Resnick & Varian, 1997)). They are one of the well established artificial intelligence applications in modern computer science, and are being used as typical software components in e-Commerce, m-Commerce (see (Sadeh, 2002)) and Tourism systems (see (Werthner & Klein, 1999)). Its mathematical foundations share a multitude of algorithms with other prominent fields, such as adaptive hypermedia and the semantic web.

Designing a recommender system application has some fundamental differences to software design for other applications. The overall system architecture depends heavily on the choice of algorithms. For recommender systems, the design phase includes several decisions at a very low application level. A failure to answer these fundamental questions will result in an inconcise system specification and will result in the problems this causes in software development: late changes to the fundamental design may become inevitable, causing a cost explosion. The specific design choices for a recommender system depend mostly on the following issues:

- The nature of the problem (as common in general software design).
- The computational resources available for the recommender component (more crucial than in general software design).
- The concise types of information available to the system (earlier and much more detailed than in general software design).

As we aim to give an insight into the crucial design choices concerning approach, algorithm and architecture of the recommender system, we will give a brief overview of the four basic approaches for such systems. These four basic approaches differ not only in the methods used, but also in the ``real-world interpretation" of the underlying algorithm. Based on the application, and the basic aims and objectives of the application, there may be reasons to opt for one approach rather than the other.

- Collaborative Filtering - These techniques are used in the earliest and most researched recommender systems (see (Breese, 1998; Resnick, 1994)). Often also referred to as *social filtering*, these algorithms focus on the behavior of users *on* items, which are to be recommended, rather than on the *internal nature* of the items themselves. The social approach is the technical means, which most closely resembles the nature of ”real-life recommendations”. We can assume that these algorithms have a semantic affinity to both the concept of collaborating individuals and the process of finding persons with similar interest.

- Content-based Filtering - Content-based systems focus on the internal nature of items, or on the content of description files (see (Mladenic, 1999; Hinton, 2000)). These systems utilize two main classes of algorithms, either from the field of information retrieval or attribute-based filtering systems. A content-based approach favors the semantics of the content over social interactions or user behavior. In some application domains, the content of an item may be crucial to every application. This means that systems with a severe focus on item content should use a content-based approach rather than a social approach (i.e. on the actual content, not on user interaction).
- Knowledge-based Filtering - These systems rely on an explicit representation of knowledge, usually as collections of statements, ontologies or other forms of rule systems (see e.g. (Aamodt & Plaza, 1994; Martinovska, 2002)). While the high performance and flexibility makes the knowledge-based approach suitable for most tasks, applications with a strong focus on content or social semantics can be realized more easily using the respective specialized approach. If an basic application requires reasoning or inference, choosing the knowledge-based approach allows the developers to benefit from the software components, knowledge representation and rules devised for the system in general.
- Hybrid Systems - Hybrid systems can merge any combination of the above methods and metrics (see (Baudisch, 1999; Popescul, 2001)). Typically, hybrid recommender systems would compute ratings (or simply ”scores”) from a number of ”internal algorithms”, before combining these in a single metric to allow consistent ranking. In some cases, the preliminary results of the internal algorithms are stored component-wise in a vector, before crafting a single-dimensional rating for ranking.

As we will discuss below, both the basic requirements, information availability and heterogeneity of the application domain make hybrid systems a suitable choice for a TRS.

2 Designing Tourism Recommender Systems

Recommender systems are now a popular research area and increasingly used for eCommerce. But travel recommender systems are still difficult to build because developers of a travel recommender system need to consider not only the specific nature of the travel decision process, but also the wide range of heterogeneous information available in this domain.

We already established that all four basic approaches to the recommending problem form a basis for an Tourism recommender system, and a hybrid approach is a candidate in this rich and heterogeneous information domain. But before we can state any preference for one or the other, we have to consider the real-world objects, and their properties (or *aspects*), which may be relevant for a TRS.

2.1 From Objects and Aspects to Algorithms

We know that there is a wide range of different attributes of tourism entities, which may be essential or desirable for a TRS (like users preferences, destinations, events, restaurants, accommodations, etc.):

- Spatio-temporal data,
- cost and financial aspects of items,
- associated media files (texts, images, audio, video),
- classification information (wellness, adventure, etc),
- ontological modeling of object specific properties (databases, ontologies, etc).

Based on our considerations about the typical types of data and information on items, we can sketch some initial considerations on what filtering techniques are suitable for a specific type of information (see Table 1). We can now examine the typical advantages and challenges of these techniques:

Table 1. Filtering Algorithms for Information Types

Type of Information	Filtering Technique
Spatio-Temporal	database selection, (fuzzy) logic
Financial	database selection and fuzzy rules
Media Classification	media classification ontological modeling
Ontological Modeling	database design and selection, ontology building and reasoning
Users	collaborative filtering

- Database design and selection - easy to develop and maintain. A database selection of items can be performed very easily and with few resources, making it a good method for an initial reduction of item lists.
- Ontologies - for larger projects, ontologies provide an efficient means of representing and storing objects-of-interest, and provide a basis for reasoning. However, the development of an ontology requires more development effort than traditional database design.
- Knowledge-based filtering and reasoning - based on the envisaged functionality, the development effort can range from low to very high. This technique allows developers to incorporate explicit domain knowledge. This can be a direct means of increasing recommender system performance without resorting to more powerful algorithms (see (Fensel, 2000) on strengthening and weakening of problem-solving methods). Depending on the number of different rules, and the complexity of the chosen form of logic, the process of rule building can be very simple, or very complex. Fuzzy rules are a special case, as they allow reasoning with numerical results. This ”soft nature” of fuzzy logic make them ideal for Tourism applications (e.g. prefer a restaurant which is within a certain distance, but without having a fixed maximum distance for the selection).

- Content-based filtering and media classification methods - allows the integration of widely available information into the system (e.g. textual descriptions of destinations, restaurants and other items-of-interests). Text classification is a widely available and thoroughly researched technology. The computational cost can range from very little to very expensive, based on the projected accuracy and functionality. Image classification is a well established technology, but rarely used in recommender systems. The computational cost is much higher than with text classification methods, but the available algorithms and techniques offer similar functionality. Audio and video material could also be included into a recommender system, but the classification of such material requires extremely high computational power, making texts and possibly also images a better choice of material.
- Collaborative filtering - requires some integrative development effort to collect user votes on items. As Tourism systems usually have a high number of items-of-interest and few user votes on items, collaborative filtering usually requires some modifications to be used in this application domain. However, there are numerous ways for computing and using user similarity in Tourism applications.
- Hybrid systems - We know that all three basic types of recommender systems (social, content-based and knowledge-based) are suitable for the Tourism domain, as we can make use of social interaction, associated media resources and decisions based on rules. Subsuming, we can say that both the complexity and the heterogeneous collections of information in this domain favor the use of hybrid systems.

The final choice on recommender system approach depends on the information sources and objects-of-interest, which are to be used in the system. Some of these sources of information are easy to obtain and maintain, others involve more cost and effort. In fact, this choice is the main determining factor of a recommender system. As we have seen, all basic recommending approaches are applicable to the Tourism domain. Moreover, the heterogeneity of this domain favors the use of hybrid recommenders, which is another indication that all approaches can produce valuable contributions to the inference process. We also know that the choice of approach and algorithm is an inductive conclusion, based on the types of information used in a system. While some forms of information may be of interest for the overall system functionality (e.g. documents on destinations and other objects-of-interest, ontological models, etc), other data may have to be acquired solemnly for use in the recommender component. This data can cause considerate additional maintenance and acquisition cost during deployment. So the final choice of recommender system approach is based on the economic criterion of ”the cost and availability of information”. As the financial parameters of a TRS are typically known prior to design and implementation, the selection of information for integration in the system can provide the basis for all further considerations, ranging from basic design to the concise choice of algorithms. Let us now consider three examples of TRS at different economic scales.

A TRS, which can be described as ”small” according to economical criteria would typically rely on information and data, which is available for free, or at a low price. Such a system could make use of the World Wide Web as an open, cheap and extensive source of information. Autonomous web crawlers could harvest documents directly from the service providers. This could be achieved through lists of hotel chains, regional tourism sites, commercial providers of tourist trips and excursions, and other open document archives. Even though the raw data would be provided by others, it could be reused as-is, and the integration of this information would also benefit the service providers, who become a ”remote part” of the TRS. Once these documents have been obtained, they can be classified for different regions, financial groups, activities and other forms of categories. User can then specify their interest in the categories provided by the system, and the system can directly supply them with the best matching documents, thus recommending the offers harvested from the web. While the implementation of such a system can be quite complex and costly, once operative, the system requires little maintenance investments. This means that the usage of freely available documents and content-based filtering techniques can lead to a cheap, but efficient system.

An economically larger system can cooperate with professional data providers to make use of in-depth descriptions of the individual items. In this case, the system does not merely use a set of categories, but could make use of a full ontological model. Such a model would not only have knowledge of categories, but also of individual items and their properties. This allows developers to include more advanced means of filtering, such as database selection, case-based filtering and rule-based filtering. If individual items are known to a system, further means of user support can be integrated, namely user profiling and interest prediction. Such features can in turn be used to implement simple means of collaborative filtering, which can increase the system's accuracy and serve as a ”community building factor”. However, the necessity to collect and input the data for individual items into the system creates a somewhat bigger maintenance effort. Needless to say, the inexpensive content-based approach of the small system example can also be integrated into this system, leading to a hybrid approach with more functionality (i.e. providing documents) and a higher accuracy.

Large TRS can include proactive means of interacting with the user. If the economical scale permits it, bonus-point, gift or specialized-offer systems can be used to persuade users to use the system for an extended time, or serve as a motivation for vote collection on individual items. This can increase the accuracy of user interest predictions, and allow the full deployment of collaborative filtering. Such a system would also require some form of knowledge model, and could also be extended using content-based filtering, should this be considered a ”valuable extension” of the system's functionality, justifying the extra development and deployment investments.

3 Designing the Recommender System Architecture

As we have seen above, we can derive (or induce) the filtering techniques from the types of information known to the system. Based on this set of techniques, designing the basic recommending architecture can be done easily. Figure 1 depicts a potential recommender component architecture sketch, which combines *all* of the above methods in a hybrid system. Reduced systems, which do not use all of these methods, could be derived by omitting all irrelevant components. As with all hybrid systems, the sequence and combination of sub-components can be varied to modify the functionality and behavior according to the computational requirements of the system. We can identify the following processing steps:

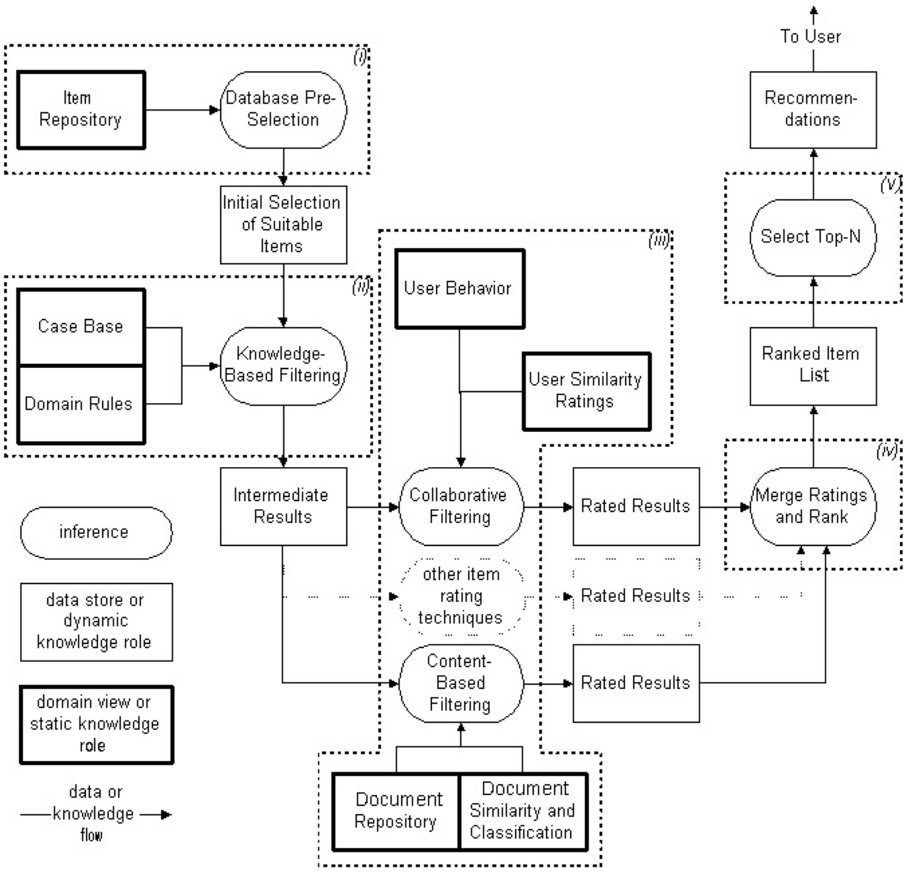

Figure 1. Architecture for a Hybrid Filtering Component

(i) Database pre-selection provides an initial selection of items, based on simple database interactions (similar to rule-based filtering using boolean logic). This yields a very efficient reduction of the number of items, which have to be processed, at a very early stage of the workflow.

(ii) Knowledge-based filtering methods can now either provide initial item ratings (using fuzzy logic), or perform a further reduction of the item set size (using boolean logic). The usage of knowledge-based filtering allows developers to make use of explicit domain knowledge.

(iii) We can make use of collaborative filtering, content-based filtering methods, or virtually any other form of item rating technique to obtain one or more numerical ratings for every item. This allows us to incorporate algorithmic or implicit domain knowledge.

(iv) If we are dealing with a hybrid system, having more than one rating (i.e. vote prediction, similarity, relevance, etc) for every item, we can now merge these multi-dimensional votes to a single value (e.g. using the mean value, multi-dimensional vote vector length, etc).

(v) By selecting the N best scoring items, we obtain a final set of recommendations.

4 Summary and Conclusion

After an initial analysis, we have concluded that we can make beneficial use of the following AI techniques in a TRS:

- Database pre-selection - Quick reduction of large amounts of data.
- Content-based scoring - Allows to make use of typical tourist media information to add scores based on content and its semantics to the overall inference process.
- Incorporating implicit domain knowledge - Allows to integrate handcrafted domain knowledge and expertise.
- Incorporating social aspects - Allows to integrate groups of users with a similar interest.
- User profiling - Representing users with respect the dynamic nature of users in tourism, as to combine all other approaches used in a system.

As we know, the Tourism application domain is based on very heterogeneous collections of information (see Section 1.1 and 1.2). An initial conclusion is that hybrid approaches to the recommending problem are very well suited, because this approach does not force a strict limitation to one type of information or data throughout the system, allowing the integration of any other filtering technique.

As all the different types of information are well suited for usage in a recommender system, but call for very different algorithms, the choice of the recommender system approaches to be used within the system (hybrid or not) is based *purely on the types of information, which are to be integrated into the system* (see Table 1).

Based on these considerations, we can sketch the workflow for the design process of an Tourism recommender system as:

- Determine types of information - apply economic criteria.
- Derive filtering techniques - consult Table 1.
- Choice and refinement of the architecture - modify Figure 1.
- Detailed AI design & implementation - specialized development teams.

Trivially, economic criteria are common throughout all commercial applications, regardless of the application domain. But for a TRS, it is the economic criterion of ”buying, creating and maintaining information and data *during deployment*”, which has a major impact throughout all levels of the system.

So even though the diverse and heterogeneous information environment in Tourism may initially cause a high cognitive complexity for developers, it is this very same heterogeneity that allow a great amount of flexibility for a TRS, regarding both functionality and ”economic scope”.

References

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

Baudisch, P. (1999). Joining collaborative and content-based filtering. In *Interacting with Recommender Systems,* Online Proceedings of the CHI '99 Workshop.

Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical Analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence* (pp. 43-52).

Delgado, J., & Davidson, R. (2002). Knowledge bases and user profiling in travel and hospitality recommender systems. In *Proceedings of the IFITT's Global Forum for Travel and Tourism Technology and eBusiness, focusing on Multi-Channel-Strategies,* ENTER 2002. Springer.

Fensel, D. (2000). *Problem-Solving Methods: Understanding, Description, Development and Reuse*. Springer.

Hinton, G. E., Ghahramani, Z., & Teh, Y. W. (2000). Learning to parse images. In Solla, S. A., Leen, T. K., & Muller, K. R., *Advances in Neural Information Processing Systems*: Vol. 12, (pp. 463-469). MIT Press, Cambridge, MA.

Martinovska, C. (2002). A fuzzy-based approach to user model refinement in adaptive hypermedia systems. In *Adaptive Hypermedia and Adaptive Web-Based Systems* (pp. 411-414). Springer.

Mladenic, D. (1999). Text-learning and related intelligent agents. *IEEE Expert Special Issue on Applications of Intelligent Information Retrieval*.

Pazzani, M. (1999). A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*.

Popescuil, A., Ungar, L., Pennock, D., & Lawrence, S. (2001). Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In 17th Conference on *Uncertainty in Artificial Intelligence*, Seattle, Washington.

PRISMA. (2001). *PRISMA – Providing Innovative Service Models and Assessment, Report on changes and trends in service delivery*. Deliverable D2.1, Information Societies Technologies (IST) Programme.

Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P., & Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. *Proceedings of the*

ACM 1994 Conference on Computer Supported Cooperative Work (pp. 175-186), Chapel Hill, North Carolina.
Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40 (3), 56-58.
Ricci F. (2002), Travel Recommender Systems. *IEEE Intelligent Systems*, November/December 2002, 55-57.
Sadeh, N. M. (2002*). m-Commerce: Technologies, Services and Business Models*. Wiley.
Smeral, E. (2003*). Die Zukunft des internationalen Tourismus, Entwicklungsperspektiven für das 21. Jahrhundert*. Linde Verlag Wien.
Werthner, H., & Klein, S. (1999). *Information Technology and Tourism, a Challenging Relationship*. Springer.