Expert Systems with Applications 41 (2014) 6796–6806

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

An efficient Recommendation System based on the Optimal Stopping Theory

Kostas Kolomvatsos [a,*], Christos Anagnostopoulos [b], Stathes Hadjiefthymiades [c]

[a] Department of Computer Science, University of Thessaly, 35100 Lamia, Greece
[b] Department of Informatics, Ionian University, 49100 Corfu, Greece
[c] Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, 15784 Athens, Greece

ARTICLE INFO

Article history:
Available online 13 May 2014

Keywords:
Recommender Systems
Optimal Stopping Theory
Quality of Recommendation
Stochastic decision making

ABSTRACT

A Recommendation System (RS) aims to deliver meaningful recommendations to users for items (e.g., music and books), which are of high interest to them. We consider an RS which directly communicates with a set of providers in order to access the information of the items (e.g., descriptions), rate them according to the user's preferences, and deliver an Item List (IL). The RS is enhanced with a mechanism, which sequentially observes the rating information (e.g., similarity degree) of the items and decides when to deliver the IL to the user, without exhausting the entire set of providers. Hence, the RS saves time and resources. We propose two mechanisms based on the theory of optimal stopping. Both mechanisms deliver an IL, which sufficiently matches to the user's needs having examined a partial set of items. That is, the number of items in the delivered IL is optimal, producing a high level of user satisfaction, i.e., Quality of Recommendation (QoR). Our simulations reveal the efficiency of the mechanisms and quantify the benefits stemming from their adoption.

1. Introduction

1.1. Motivation

A *Recommendation System* (RS) produces individualized recommendations, i.e., a list of items (e.g., products or services), to a user based on her preferences (Burke, 2002). Providers possess a finite number of items in their property and attempt to offer the requested items' information to the users through the RS. An RS is based on information related to: (a) the user preferences, (b) the description of each item, and (c) the preferences of other users. The user pre-defines her personal preferences in order to affect the delivered *Item List* (IL). The RS, after receiving a request from a certain user and for a specific item description, outputs an IL, in which, the corresponding items best match the user's request.

Normally, a user defines her preferences through rate values (ratings) over items. A rating depicts the *attitude* of the user to obtain a certain item. The RS combines ratings defined by different users in order to achieve high *Quality of Recommendation* (QoR). The QoR could be derived in many ways mainly depicting the accuracy of the recommendation. In this paper, we consider that QoR depicts the matching degree of the provided IL with the user's needs. Certain approaches have been proposed for RSs. Content-based models analyze users' profiles and items' description to identify items that are of particular interest to a user (Pazzani & Billsus, 2007). They recommend items similar to those that match the user profile and those the user liked in the past. The high level architecture and an extensive survey on content-based recommender systems can be found in Lops, de Gemmis, and Semeraro (2011). An RS could result in very large ILs (in number of recommended items). A *large* IL is not efficient to be further processed by a user. Users want only the items that best match their needs. Moreover, the complexity of the solutions found in the literature for content-based recommendations probably makes their adoption to real scenarios quite difficult. Most content-based methods attempt to learn a user's model through the adoption of decision trees, ontological similarity techniques, relevance feedback algorithms, linear classifiers, and probabilistic methods. Decision trees become very large and complex, when the number of historical data becomes large as well. Relevance feedback methodologies require a 'discussion' phase with users, while machine learning approaches require the appropriate training data. Probabilistic methods are usually based on the Bayes rule and, thus, require the calculation of a priori probabilities. In addition, ontology-based approaches have the problem of heterogeneity as providers probably make use of their own ontologies/taxonomies for describing

* Corresponding author. Tel.: +30 6976220574.
E-mail addresses: kolomvatsos@cs.uth.gr (K. Kolomvatsos), christos@ionio.gr (C. Anagnostopoulos), shadj@di.uoa.gr (S. Hadjiefthymiades).

http://dx.doi.org/10.1016/j.eswa.2014.04.039
0957-4174/

and representing items. The problem in such methods is the requirement of the users' intervention (especially when their interests change) and/or the training data.

The collaborative filtering methods are categorized, according to Khabbaz and Lakshmanan (2011), as: (i) user-based, (ii) item-based, (iii) model-based, and, (iv) fusion-based approaches. In user-based approaches (Herlocker, Konstan, Borchers, & Riedl, 1999), a similarity matrix is adopted to store ratings of each user for every item. The main problem is the missing values substitution of the items description. The item-based methods (Deshpande & Karypis, 2004; Wang, de Vries, & Reinders, 2006) attempt to overcome the problems appeared in the user-based methods. Pairwise item similarities are more reliable than user similarities, thus, resulting in higher QoR. The model-based methods (Das, Datar, Garg, & Rajaram, 2007) exploit the sparsity of the data in the similarity matrix. Training examples are used to generate the appropriate model parameters. Based on such parameters, prediction methods could substitute the missing values. However, tuning a significant number of parameters has prevented these methods from practical use (Jambor & Wang, 2010). The fusion-based methods (Oku & Hattori, 2011) adopt information fusion and aggregation techniques for building the IL. They aggregate values, which represent recommendation results, features of user preferences, or correlations between user preferences and items.

We consider a content-based RS to solve some of the above mentioned problems. Content-based RSs are more easily handled and do not suffer from the lack of information. For example, they can return a result even if no information about other users ratings is present (this is a common problem in collaborative filtering methods). We focus on the case where the RS has direct interactions with a finite set of providers P, thus, exploiting the up-to-date items information. We propose two mechanisms based on the *Optimal Stopping Theory* (OST) (Peskir & Shiryaev, 2006), which deliver an IL taking into consideration the available resources and the required time for building the IL. That is, the mechanisms deliver an IL where the number of items is optimal in order to produce a high level of user satisfaction, i.e., QoR. The first mechanism, hereinafter referred to as *IND* (INDependent), is independent of the number of the providers $N = |P|$, while the second mechanism, referred to as *DEP* (DEPendent), takes into account the value of N. The enhanced RS (by adopting the proposed mechanisms) sequentially receives information about the items from P, and decides *when* it is the appropriate time to deliver the IL to the user, without exhausting the entire set P (if possible). The proposed mechanisms depend on an OST-based reward maximization technique, which corresponds to the maximum expected QoR, $E[\text{QoR}]$, of the user. Based on the OST, the RS identifies when the reward (i.e., QoR) is maximized. At that time, the RS stops retrieving item information and delivers the IL to the user. The proposed mechanisms:

- Do not require any complex modeling or any specific knowledge, thus, saving resources to the RS.
- Maximize the expected value of the recommendation (E[QoR]).
- Do not require any training process, information related to other users, or user intervention.

1.2. Related work

From early of 90s, RSs are an active research area. Many commercial and research frameworks have been proposed. In such frameworks, a variety of algorithms and techniques were adopted to provide efficient results. Specific research efforts tried to reveal that the embedding of the algorithm in the user experience dramatically affects the value to the user (Konstan & Riedl, 2012).

In this paper, we focus on a content-based RS. The adoption of a content-based RS has a number of advantages and disadvantages (Lops et al., 2011). Content-based approaches require ratings made by the user herself in contrast to collaborative filtering models that cannot derive an efficient result without other users' ratings. Furthermore, explanations on the final result could be given in terms of items descriptions and users profiles. In collaborative filtering approaches, the provided results are based on the similar tastes of users and their combination. New items are handled easier than in other models as the recommendation is based on their descriptions even if ratings are not yet present. On the other hand, content-based models depend on the performance of the content analysis methodology they utilize. The performance of the matching process of the items descriptions with the users' profiles is, in addition, a very significant issue.

A number of keyword-based recommender systems have been proposed over the past years. The most known of them are described in Asnicar and Tasso (1997), Chen and Sycara (1998), Mladenic (1999) and Moukas (1997). In such systems, every document is represented by a multi-dimensional vector where each dimension describes a specific term. Weights are devoted to define the association between documents and terms. User profiles and items description play the role of documents and terms. Vector based similarity algorithms are used in such cases.

Semantic techniques have been also used in RSs (Basile, de Gemmis, Gentile, Iaquinta, & Lops, 2007; De Gemmis, Lops, & Semeraro, 2007; Eirinaki, Vazirgiannis, & Varlamis, 2003; Magnini & Strapparava, 2001; Middleton, Shadbolt, & De Roure, 2004). The aim is to provide means for reasoning in the recommendation mechanisms. Ontologies play an important role to that, however, one can identify the problem of (semantic) heterogeneity. Different item providers could utilize different ontologies, thus, the reasoning process becomes very hard. Ontological terms are used for the semantic annotation of items and users information or to define relationships between keywords. Moreover, word disambiguation is used for mapping keywords to widely taxonomies like WordNet.[1] Such taxonomies 'translate' the content of an item and to support annotation.

Probabilistic methods are used for creating models adopting historical data (Billsus & Pazzani, 1999; Billsus & Pazzani, 2000; Mooney & Roy, 2000; Pazzani & Billsus, 1997). Such models result posterior probabilities by analyzing past data and are based on a priori probabilities. Such probabilities are related to the relationship between documents and terms (i.e., user profiles and item characteristics). Usually, the Bayes rule is the key methodology for the calculation of such probabilities while the Naive Bayes classifier is recognized by researchers as the method with the best performance (Lewis & Ringuette, 1994). The disadvantages of probabilistic models are that they require some a priori probabilities to be defined and that they violate the conditional independence of data.

In this paper, we propose two novel recommendation models with two goals:

- To maximize the expected QoR. The QoR represents the satisfaction that a user gains from the derived IL.
- To save resources and time. In real scenarios, an RS serves a large number of users. Therefore, a methodology that results the final IL in minimum time is necessary. Based on such methodology, the RS increases its performance.

Our view is different compared to the rest research efforts in the domain. We focus on the performance of the RS when serving a large number of users and *not* in the matching process. Our proposed models try to keep the QoR in high levels while searching the appropriate time to stop the process and, thus, they save

[1] <http://wordnet.princeton.edu/>.