Exploiting Relationship Among Cases to Make the Best Use of Users Feedback

Anbarasu Sekar

Indian Institute of Technology Madras, Chennai 600036, India
anbuforever@gmail.com

Abstract. Recommender systems (RSs) are built with the aim to reduce the cognitive load on the user. An efficient RS should ensure that a user spends minimal time in the process. Conversational Case-Based Recommender systems (CCBR-RSs) depend on the feedback provided by the user to learn about the preferences of the user. In our work, we exploit the relationship among the cases/products in addition to the feedback (preference-based feedback (PBF)) provided by the user in several ways to develop an efficient CCBR-RS.

1 Introduction

Imagine a situation where a user enters a shop to buy a Camera. How do we expect shopping to happen? The probability that one keeps buying cameras periodically from the same shop is very low. With no information about the preferences of the user, how do we expect the sales executive to understand the needs of the user and provide the right recommendations? Collaborative systems render little help in situations as the one explained above. This is traditionally called the cold-start condition [1]. Knowledge-based recommender systems excel where collaborative systems fail. In Conversational Case-Based Recommender Systems (CCBR-RSs), the customer/user starts out with mentioning the values of few features of the product and the system starts conversing with the user just like a sales executive and helps the user identify her product of interest. A CCBR-RS involves the user in two phases of interaction, one, recommendation phase, where the system recommends products to the user based on perceived preferences of the user, two, feedback phase, where the user gives feedback about the recommendation made to her. In PBF, the user points out one of the products from the recommended set of products as her preference.

In a domain like 'Camera', the features of the products can be ordered based on criteria outlined by McSherry [2]. For example, if we have two cameras with similar feature values except one camera has better resolution than the other, with high probability people tend to choose the camera with higher resolution. Hence the feature *resolution* of the camera is categorized as More is Better (MIB) feature. Similarly, people tend to choose products with lesser *price* among two products with similar feature values categorizing *price* into Less is Better (LIB) feature. We use these categorizations to identify the relationship among products and use them to make the best use of feedback provided by the user. The aim of our work is to reduce the cognitive load on the user by reducing the conversation length. Our work has three broad goals, (a) We deal with profiling products based on the feature weights under which a particular product dominates its competitors and use it to enrich the measure of utility, (b) The second work identifies the trade-off relationship among products and helps users identify products that make trade-offs that are in line with their preferences and (c) We aggregate evidence for each product in the domain through the feedback provided by the user, the feedback provided on a set of products is propagated to every product in the domain based on its relationship with them. In each of these works, we show how our methods help to boost the performance of the system by empirically evaluating it on three datasets [11].

2 Research Progress

2.1 Utility based on dominance relation among products

Status: Published. "Towards Bridging the Gap between Manufacturer and Users to Facilitate Better Recommendation" [6]

The work by Mouli et al. [4] points out the drawback in learning weights of the features without considering the dependency among features. For example among a set of cameras shown to the user, if the user prefers a product with the least *zoom*, it doesn't mean the importance given to *zoom* is the least but it may also be the reason that the user rejected products are costlier. The authors use a Multi-Attribute Utility Theory (MAUT) [5] based formulation to capture the utility of a product. A fair assumption that the utility of the product preferred by the user is higher or equal to every other product rejected by the user in a given interaction cycle is made. The above assumption allows us to formulate inequalities that can be solved for the feature weights. In a given interaction cycle the number of inequalities one can derive would be far less than the number of features in the domain and hence the solutions to the inequalities would be a convex region of vectors in the feature weights space, let us term the region, *preference region of the user*. A single weight vector from the *preference region* is selected as the feature weights to compute weighted similarity. The selection of the weight vector is based on the assumption that the preferences of the user does not change drastically in successive interaction cycles. So a weight vector that is closest to the feature weights vector assumed in the previous interaction cycle is selected. The method is termed Compromise Driven Preference Model (CDPM) [4].

As the first part of our doctoral research, we enrich the utility of the weighted similarity measure (feature weights determined based on CDPM) with additional information from the domain. We call this additional information dominance knowledge. We argue that each product has prospective buyers and the importance that the buyers of a particular product would give to the features of that product would be similar. We make an assumption that the products that are similar to each other are competitors to each other and a prospective buyer of a particular product would prefer it over its competitors. Similar to CDPM we can derive inequalities by comparing every product with its competitors. The solutions to the feature weights are contained in a convex region in the feature weights space. We term this region *dominance region of the product*. Under the feature weights assumptions from the dominance region of a particular product, the utility of that product is greater or equal to all its competitors. Each product has its own *dominance region* and each user has their *preference region*. The dominance region characterises the prospective buyer of the product in terms of feature weights. We define a measure of overlap between the dominance region of a certain product and the preference region of the user. We combine the overlap score of a product and its weighted similarity to get the enriched utility of the product. We term our method Predicted Preference Region based Method (PPRM).

2.2 Utility based on trade-off relations among products

Status: Published. "Why Did Naethan Pick Android over Apple? Exploiting Trade-offs in Learning User Preferences" [8]

Given a user query and a product, we can determine the compromises the users have to make in accepting the product. McSherry [3] in his work shows how the success of the recommender system can be improved by offering the user all possible compromises available in the domain as the compromise a user would make is not known in advance. Compromise is a relationship between the user query and a product. We argue in our work that trade-offs are the relationship among products and can be determined even before the user's query is encountered. Given a pair of cars, one a high-performance car and another a commuter car, we can determine what features dominate in each of the cars. Irrespective of the query, if the two cars are shown to the user and if the user prefers the commuter car, then we can identify the set of features the user prefers over the other set of features. The dominance relation is based on MIB and LIB categorization of the features. The user who prefers a commuter car will mostly prefer the feature *Fuel efficiency* over *Top Speed*.

In our work, we define a representation for trade-offs and a similarity measure between a pair of trade-offs. In each interaction cycle, we get the PBF from the user. We compare the PBF with each of the rejected products and determine the trade-offs the user preferred product makes with each of the rejected products. In the next interaction cycle, the usefulness of a product is determined by its similarity with the PBF and the similarity between the trade-offs with the user rejected products. For a product to be included in the recommendation set its similarity with the PBF should be high and the trade-offs it makes with the rejected products should be similar to the trade-offs the PBF makes with the rejected products.

2.3 From PBF to Evidences

Status: Accepted. "Show me your friends, I'll tell you who you are: Recommending products based on hidden evidence" [9]

Published. “Exploiting the interplay among products for efficient recommendations” [10]

We propose a novel view to the process of conversation. The feedback in each interaction cycle is used for aggregating evidence for each product in the domain. The evidence could be both positive and negative. We try to model a sales executive who associates the preferences expressed by a user with some product in the domain and recommends the same to the user. The relation among products both in terms of its MAUT based similarity and its similarity based on trade-off is exploited to propagate positive and negative evidences. In each interaction cycle, one product is preferred and the rest are rejected. A product gets positive evidence proportional to its similarity to PBF and negative evidence proportional to its similarity with the rejected products. We categorize the evidence into dominance based evidence and trade-off based evidence. We discuss how the works from literature can be viewed in the light of evidence-based recommendation and suggest ways in which the propagation of noise in the evidence can be minimised.

In our latest work in [10] we incorporated higher-order evidence mining based on the relationship among products. We proposed a way of constructing precedence graph with the products in the domain as the nodes and the edge from one product to another denoting the precedence of one product over the other with the weights implying the strength of evidence. We used the random surfer model to compute the stationary distribution of the products(nodes in the graph) and use it as a measure of utility.

2.4 Evaluation and Results

We evaluate the efficiency of our methods based on leave-one-out methodology [7]. The product that is most similar to the left out product is set as the target. We assume an artificial agent that gives PBF in every interaction cycle. In each recommendation phase, the PBF choice is decided based on similarity to the left out product. We simulate easy, medium and hard conditions by randomly picking feature values of different sizes from the left out product as the initial query. The sizes chosen are 5, 3 and 1 respective to easy, medium and hard conditions. Instead of leaving out every product we randomly pick one product from the domain and measure the number of cycles it takes for the three query sizes. We do this process for 1000 times and the average cycle length for each of the query sizes is used as the metric for comparison against several methods. We split the 1000 queries into 10 folds and check for statistical significance of the difference in average across methods.

We have found our methods to have significantly better performance supporting our hypotheses. The best performance is recorded for the Used Car dataset. The average reduction in the number of cycles compared to the baseline work for Used Car dataset for the three methods is as follows. PPRM [6] showed 40, 43 and 40 percent reduction in average number of cycles for hard, medium and easy queries respectively. Similarly our trade-off based method [8] showed 54, 49 and 43 percent reduction and the Evidence based method showed 61, 60 and

61 percent reduction in the average number of cycles compared to the vanilla similarity based CCBR-RS.

3 Future Works

The advantage with CCBR-RSs is its ability to explain why a recommendation is made to the user. We would like to explore the task of explaining recommendations in our approaches.

References

1. Gediminas Adomavicius, Alexander Tuzhilin: 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, v.17 n.6, pp.734-749 (2005)
2. McSherry, David.: Coverage-Optimized Retrieval. Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, pages 1349-1350. Morgan Kaufmann Publishers Inc. (2003)
3. McSherry, David.: Similarity and compromise. International Conference on Case-Based Reasoning. Springer Berlin Heidelberg (2003)
4. Mouli, S. Chandra, and Sutanu Chakraborti.: Making the Most of Preference Feedback by Modeling Feature Dependencies. Proceedings of the 9th ACM Conference on Recommender Systems. ACM (2015)
5. Keeney, Ralph L., and Howard Raiffa.: Decisions with multiple objectives: preferences and value trade-offs. Cambridge university press (1993)
6. Anbarasu Sekar, and Sutanu Chakraborti. Towards Bridging the Gap between Manufacturer and Users to Facilitate Better Recommendation. The Thirty-First International Flairs Conference. (2018)
7. McGinty L. and Smyth B.: Evaluating preference-based feedback in recommender systems. In Artificial Intelligence and Cognitive Science, pp 209-214. Springer (2002)
8. Sekar A., Ganesan D., Chakraborti S. Why Did Naethan Pick Android over Apple? Exploiting Trade-offs in Learning User Preferences. In: Cox M., Funk P., Begum S. (eds) Case-Based Reasoning Research and Development. ICCBR 2018. Lecture Notes in Computer Science, v. 11156. Springer (2018)
9. Anbarasu Sekar and Sutanu Chakraborti.: Show me your friends, I'll tell you who you are: Recommending products based on hidden evidence. In 27^{th} International Conference on Case-Based Reasoning, Otzenhausen, (2019) (Accepted)
10. Anbarasu Sekar and Sutanu Chakraborti.: Exploiting the interplay among products for efficient recommendations. In International Journal of Advanced Computer Science and Applications, Volume: 10 Issue: 6, ISSN 2158-107X (Print), pp.549-556, The Science and Information (SAI) Organization Limited, (2019)
11. http://www.mycbr-project.net/download.html