IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Volume-03, Issue-3 E-ISSN: 2347-2693

# A Survey on Alleviating Cold – Start Problem in Lars* Using Hybrid System

Mili Mohan[1*] and Robert.S[2]

[1,2] Department of CSE, Marian Engineering College, Thiruvananthapuram,India

www.ijcseonline.org

Received: Feb/20/2015 Revised: Mar/01/2015 Accepted: Mar/18/2015 Published: Mar/31/2015

**Abstract-** Number of people who uses internet and websites for various purposes is increasing at an astonishing rate. More and more people rely on online sites for purchasing rented movies, songs, apparels, books etc. The competition between numbers of sites forced the web site owners to provide personalized services to their customers. So the recommender systems came into existence. LARS* is a location-aware recommender system that uses location based ratings to produce recommendations. LARS* supports a taxonomy of three novel classes of location-based ratings, namely, spatial ratings for non-spatial items, non-spatial ratings for spatial items, and spatial ratings for spatial items. The item based collaborative filtering used for generating recommendations in LARS* suffers from cold start problem. In cold start problem, the recommenders cannot draw inferences for users who are new to the system (new user problem) and for items which does not have sufficient ratings (new item problem). New user cold start problem can be resolved by utilizing the demographic data explicitly given by a user. Also the content based filtering does not suffer from new item cold start problem. From the survey carried out, a hybrid recommender system which exploits the demographic and content based filtering features can be used for alleviating cold start problem.

**Keywords:** Location Aware Recommender System, Collaborative filtering, cold-start problem, demographic filtering, content based filtering, Hybrid Systems

## I INTRODUCTION

With the rapid advancements in the field of position localization techniques, people are allowed to share their locations and location related contents through different social networking sites. Location data bridges the gap between the physical and digital worlds and enables a deeper understanding of user preferences and behavior. This addition of vast geo-spatial datasets has stimulated research into novel recommender systems that seek to facilitate user's travels and social interactions. Recently, advances in location-acquisition and wireless communication technologies have enabled the creation of location-based social networking services, such as Foursquare, MovieLens etc. In such a service, users can easily share their geospatial locations and location related content in the physical world via online platforms. For example, a user with a mobile phone can share comments with his social network about a restaurant at which he has dined on an online social site.

LARS*[1] is a location aware recommender system which is built specifically to generate high quality location based recommendations. It is a single framework consisting of three types of location based ratings:

1. Spatial ratings for non-spatial items represented as a 4-tuple (user,ulocation,rating,item).
2. Non-spatial ratings for spatial items represented as a 4-tuple (user,rating,item,ilocation)
3. Spatial rating for spatial items which is represented as a 5-tuple (user,ulocation,rating,item,ilocation)

Here ulocation & ilocation represents the user location and item location respectively.

LARS*[1] produces recommendations using spatial ratings for non-spatial items, i.e., the tuple (user, ulocation, rating, item) by employing a user partitioning technique that exploits preference locality. This technique uses an adaptive pyramid structure to partition ratings by their user location attribute into spatial regions of varying sizes at different hierarchies. For a querying user located in a region R, it applies an existing collaborative filtering technique [3] that utilizes only the ratings located in R. It produces recommendations using non-spatial ratings for spatial items, i.e., the tuple (user, rating, item, ilocation) by using travel penalty, a technique that exploits travel locality. This technique penalizes recommendation candidates the further they are in travel distance to a querying user. To produce recommendations using spatial ratings for spatial items, i.e., the tuple (user, ulocation, rating, item, ilocation).LARS* employs both the user partitioning and travel penalty techniques to address the user and item locations associated with the ratings. This is a salient feature of LARS*, as the two techniques can be used separately, or in concert, depending on the location-based rating type available in the system.

## II RELATED WORKS

Recommender systems are nowadays used in a large variety of application setting, ranging from online stores, music and movie recommendation, to social media recommender and many more. Each of these applications has its particular characteristics, with greatly differing temporal dynamics or volatility, amounts of available data, use of explicit or implicit indicators, etc. Recommendation algorithms are best known for their use on e-commerce Web sites, where they use input about a customer's interests to generate a list of recommended items. Amazon.com [3] uses recommendations as a targeted marketing tool in many email campaigns and on most of its Web sites' pages, including the high traffic Amazon.com homepage. Clicking on the "Your Recommendations" link leads customers to an area where they can filter their recommendations by product line and subject area, rate the recommended products, rate their previous purchases, and see why items are recommended (Figure 1).

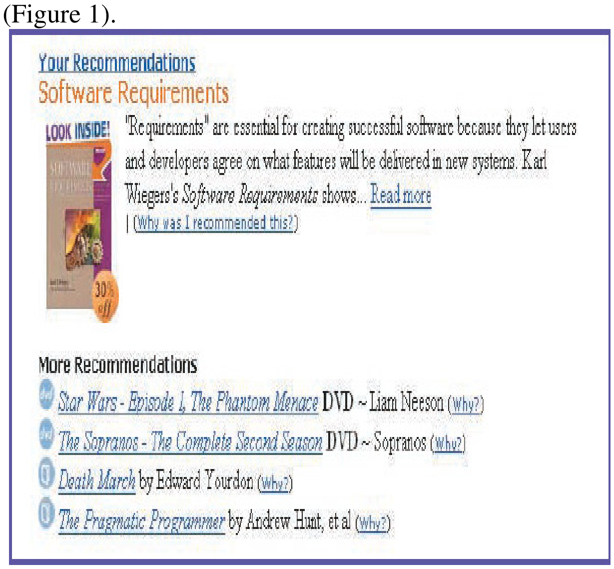


Figure 1. The "Your Recommendations" feature on the Amazon.com homepage.

The demographic-based and collaborative filtering approaches hybridization had been introduced by researches for improving the recommendation quality rather than solving "cold-start problem". A group of researchers have applied a hybrid model-based approach on movie domain using user demographic data to enhance the recommendation suggestion process, it classified the genres of movies based on user demographic attributes, such as user age (kid, teenager or adult), student (yes or no), have children (yes or no) and gender (female or male). Additionally, other researchers modified user similarity calculation method to employ the hybridization of demographic and collaborative approaches. A modification to k-nearest neighborhood had been introduced which calculates the similarity scores between the target user and other users forming a neighborhood, increasing the scores of users having similar ratings and demographic attribute (each demographic attribute had been evaluated along similar ratings separately). Whereas another research work demonstrated another modified version of k-nearest neighborhood by adding a user demographic vector to the user profile, the similarity calculation consider both ratings and demographic vector (holding all of the demographic attributes).

### A. *Motivation*

Even though LARS* is a novel framework for generating location based ratings for spatial items and user, it suffers from cold start problem as LARS* is using an item based collaborative filtering technique for generating recommendations. The cold start problem arises when a user or a content item does not have sufficient historical data known to the system (or none at all), which makes it impossible to recommend content for new users or to recommend new offers.

This paper is organized as follows: Section III gives an overview of different recommendation techniques. Section IV explains about hybridization of different recommendation techniques. Section V concludes the paper.

## III RECOMMENDATION TECHNIQUES

This section provides an overview of different recommendation techniques.

### A. *Collaborative Filtering*

Collaborative filtering (CF) [3] [4] assumes a set of n users and a set of m items. Each user expresses opinions about a set of items. These ratings can be either numeric or unary and are represented as a matrix with users and items as dimensions. CF [1] generates top-k recommendations by applying cosine similarity or any other similarity computing mechanism on these matrixes. For that a similarity score is computed for each item that has at least one common rating by same user. For instance, if you have rated a certain set of items particularly highly, the system looks for other people who have also rated those items highly and recommends to you those items that they have also rated highly that you have not rated. These recommender algorithms are based on the idea of: "people who liked those things you said you liked also liked these extra things – maybe you will too". The exact details of how the algorithms work vary, but the approach is similar. It is robust, because the algorithm does not need to know anything about the nature of what is being recommended (movies, books, music, etc.). It just looks for patterns amongst people's ratings. A detailed survey of collaborative filtering systems provided three main approaches:

1. *Memory-based collaborative filtering*. Based solely on user preferences these techniques need not consider the content of documents They typically compute similarity between items or users (using methods such as Pearson correlation and vector cosine similarity) and then make a prediction for a particular user (using weighted averages of other users' ratings). They can have problems generating recommendations involving new users or items lacking any ratings (the 'cold start' problem).

2. *Model-based collaborative filtering*. An alternative approach to using the user-item ratings is to allow a model to learn from the data and then use just that model to make the predictions.

A variety of learning algorithms have been used to generate the model, including Bayesian and clustering approaches.

*3.Hybrids*. Combining content-based and preference-based approaches to compensate for the weaknesses of each method. For example, where pure preference-based systems can struggle with items that have not been rated (or users who have not made ratings), this is precisely where a content-based technique (such as full-text querying) performs well.

In LARS*, item based CF is used and is sufficient but it suffers from the cold-start problem [6], which arises when the user or item has no historical data to the system.

### B. *Demographic Filtering*

The demographic filtering [9] generate recommendations based on the demographic profile (age, job, gender etc) of the user. This technique uses a co-relation between people like collaborative ones but using different items. The demographic-based recommendation process performs three stages: data input, similarity calculation and recommendation calculation (as shown in Fig. 2). Data input is the stage which holds new target user's demographic data (the user who requires recommendations) and also ratings and demographic data of the rest of users. Similarity calculation stage utilizes users' demographic data to obtain a number of users having similar demographic data to the target user forming a neighborhood. Finally,

Recommendation calculation stage obtains items which have been commonly positive-rated by neighborhood users to be suggested to the target user. Furthermore, the similarity calculation stage requires selecting the demographic attributes to be used for calculating the similarities. The advantage of this technique is that it doesn't need a history of user item as in collaborative ad content based filtering. This recommender obtains group of user having similar demographic attribute(s) forming a neighborhood from which newly recommended items are generated as discussed in [9]. So this technique can be used in alleviation of new user cold start problem aroused due to the collaborative filtering in LARS*.

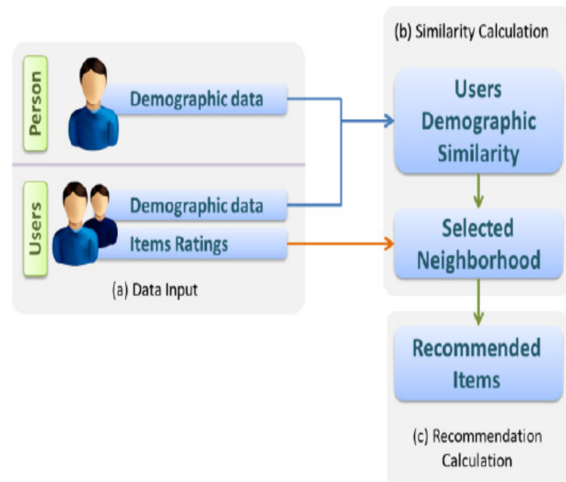


Figure. 2. Demographic-based approach for new users.

### C. *Content based Filtering*

Content based filtering [7] generate recommendations based on the utility of the item for the user which is estimated by utilities that are assigned by the user for the similar item. Systems implementing a content-based recommendation approach analyze a set of documents and/or descriptions of items previously rated by a user, and build a model or profile of user interests based on the features of the objects rated by that user. The profile is a structured representation of user interests, adopted to recommend new interesting items. The recommendation process basically consists in matching up the attributes of the user profile against the attributes of a content object. The result is a relevance judgment that represents the user's level of interest in that object. If a profile accurately reflects user preferences, it is of tremendous advantage for the effectiveness of an information access process. For instance, it could be used to filter search results by deciding whether a user is interested in a specific Web page or not and, in the negative case, preventing it from being displayed.

Content-based Information Filtering (IF) systems need proper techniques for representing the items and producing the user profile, and some strategies for comparing the user profile with the item representation. The high level architecture of a content based recommender system is depicted in Figure 3. The recommendation process is performed in three steps, each of which is handled by a separate component:

 

• CONTENT ANALYZER – When information has no structure (e.g. text), some kind of pre-processing step is needed to extract structured relevant information. The main responsibility of the component is to represent the content of items (e.g. documents, Web pages, news, product descriptions, etc.) coming from information sources in a form suitable for the next processing steps. Data items are analyzed by feature extraction techniques in order to shift item representation from the original information space to the target one (e.g: Web pages represented as keyword vectors). This representation is the input to the PROFILE LEARNER and FILTERING COMPONENT;

• PROFILE LEARNER – This module collects data representative of the user preferences and tries to generalize this data, in order to construct the user profile. Usually, the generalization strategy is realized through machine learning techniques, which are able to infer a model of user interests starting from items liked or disliked in the past. For instance, the PROFILE LEARNER of a Web page recommender can implement a relevance feedback method in which the learning technique combines vectors of positive and negative examples into a prototype vector representing the user profile. Training examples are Web pages on which a positive or negative feedback has been provided by the user;

• FILTERING COMPONENT – This module exploits the user profile to suggest relevant items by matching the profile representation against that of items to be recommended. The result is a binary or continuous relevance judgment (computed using some similarity metrics), the latter case resulting in a ranked list of potentially interesting items. In the above mentioned example, the matching is realized by computing the cosine similarity between the prototype vector and the item vectors.

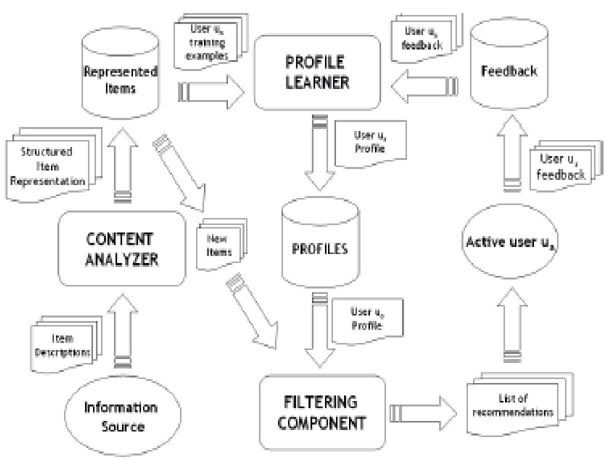


Figure 3: High level architecture of a Content-based Recommender

The first step of the recommendation process is the one performed by the CONTENT ANALYZER that usually borrows techniques from Information Retrieval systems. Item descriptions coming from *Information Source* are processed by the CONTENT ANALYZER, that extracts features (keywords, n-grams, concepts, . . . ) from unstructured text to produce a structured item representation, stored in the repository *Represented Items*. In order to construct and update the *profile* of the *active user ua* (user for which recommendations must be provided) her reactions to items are collected in some way and recorded in the repository *Feedback*. These reactions, called *annotations* or *feedback*, together with the related item descriptions, are exploited during the process of learning a model useful to predict the actual relevance of newly presented items. Users can also explicitly define their areas of interest as an initial profile without providing any feedback

As content based recommender are capable of recommending items that are not yet rated by any user, the problem of new rater/items in cold start doesn't arises. But the collaborative recommenders solely rely on the user preferences for recommendations i.e., a new item is recommended only if it is rated by sufficient number of users. So using content based filtering, new item ramp up problem can be removed. One of the drawbacks of this technique is that it can't provide suitable suggestions if the analyzed content does not contain enough information to discriminate items the user likes from items the user does not like.

### *D. Comparison of Recommendation Techniques*

This section compares the different recommendation techniques mentioned in section 3. The advantage and disadvantage of different recommendation are shown in Table 1.

## IV HYBRID SYSTEMS

In this system [10], different independent recommendation mechanisms are hybridized together to remove the limitations of the individual techniques. Commonly collaborative is combined with content based system [5] to build an efficient recommender system having better performance. There are different techniques which can be used in hybridization as explained in [11], [12], summarized as:

- Weighted : The ratings of several recommendation techniques are combined together to produce a single recommendation
- Switching: The system switches between recommendation techniques depending on the current situation

| Technique | Advantages | Disadvantages |
|---|---|---|
| **Collaborative Filtering** | A. Can identify cross-genre niches.<br>B. Domain knowledge not needed.<br>C. Adaptive: quality improves over time.<br>D. Implicit feedback sufficient | I. New user ramp-up problem<br>J. New item ramp-up problem<br>K. Gray sheep problem<br>L. Quality dependent on large historical data set.<br>M. Stability vs. plasticity problem |
| **Content Based Filtering** | B, C, D | I, L, M |
| **Demographic Filtering** | A, B, C | I, K, L, M,<br>N. Must gather demographic information |
| **KnowledgeBased Recommendation** | E. No ramp-up required<br>F. Sensitive to changes of preference<br>G. Can include non-product features<br>H. Can map from user needs to products | O. Suggestion ability static (does not learn)<br>P. Knowledge engineering required |

**Table1.** Comparison of different Recommendation Techniques

- Mixed : Recommendations from several different recommenders are presented at the same time
- Feature combination : Features from different recommendation data sources are thrown together into a single recommendation algorithm
- Cascade : One recommender refines the recommendations given by another
- Feature augmentation : Output from one technique is used as an input feature to another
- Meta-level : The model learned by one recommender is used as input to another

Here in order to remove cold start problem in LARS*, a combination of the demographic and content based techniques can be used as these remove new user problem and new item problem respectively and individually[8].

## V FUTURE WORK & CONCLUSIONS

Even though LARS* is a novel location aware recommender system which exploits the location data of user and item to generate recommendations, it has a drawback. The item based collaborative filtering used in LARS* suffers from cold start problem. So LARS* won't be able to produce accurate and efficient recommendations for a new user entering the system and also won't be able recommend the recently added and least rated items to the system users. From the survey done on different recommendation techniques it can be concluded that a system which is using a hybridization of demographic filtering (removes new user problem) and content based technique (removes new item problem) instead of collaborative filtering can be used to remove cold start problem. There by improving the efficiency and accuracy of generating top recommendations for the user.

## REFERENCES

[1] Mohamed Sarwat, Justin J. Levandoski, Ahmed Eldawy, and Mohamed F. Mokbel, "LARS*: An Effcient and Scalable Location- Aware Recommender System, "IEEE Transactions On Knowledge And Data Engineering, Vol. 26, No. 6, June 2014

[2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734-749, Jun. 2005.

[3] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," IEEE Internet Comput., vol. 7, no. 1, pp. 76{80, Jan./Feb. 2003.

[4] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms,"in Proc. Int. Conf. WWW, Hong Kong, China, 2001.

[5] James Salter and Nick Antonopoulos,"CinemaScreen Recommender Agent: Combining Collaborative and Content-Based Filtering "in IEEE Computer Society, jan/feb ,2006.

  

[6] Eduardo Castillejo and Aitor Almeida and Diego Lopez-de-Ipina , “Social Network Analysis Applied to Recommendation Systems: Alleviating the Cold-User Problem," in Ubiquitous Computing and Ambient Intelligence,Lecture Notes in Computer Science Volume 7656, 2012, pp 306-313

[7] Pasquale Lops, Marco de Gemmis and Giovanni Semeraro, “Content-based Recommender Systems: State of the Art and Trends, "in Springer Science+Business Media, LLC 2011

[8] M. Pazzani, “A Framework for Collaborative, Content-Based, and Demographic Filtering,"Arti_cial Intelligence Rev., pp. 393-408, Dec. 1999.

[9] Laila Safoury and Akram Salah, “Exploiting User Demographic Attributes for Solving Cold- Start Problem in Recommender System," Lecture Notes on Software Engineering, Vol. 1, No. 3, August 2013

[10] Robin Burke, “Hybrid Recommender Systems: Survey and Experiments,"1997

[11] Jens Grivolla, Toni Badia,Diego Campo, Miquel Sonsona, Jose-Miguel Pulido, “A hybrid recommender combining user, item and interaction data,"European Union's Seventh Framework Programme managed by REA-Research Executive Agency http://ec.europa.eu/research/rea

[12] Chris Anderson, “Recommender systems for e-shops, "Business Mathematics and Informaticspaper,2011