**Combination of multivariate logistic regression model and mining frequent itemsets in context-aware collaborative filtering**

**Abstract**

**1. Introduction**

Recent researches on collaborative filtering (CF) focus on inherent information about users and items and how to recommend such relevant items to such users. Database used to build up CF algorithms is in form of rating matrix composed of ratings that users give to items. Additional contextual factors such as time, place, condition, situation existing in real world are not considered in CF algorithms. For instance, if a user prefers to watch news program in morning and movies in evening then contextual information, namely temporary information, should be aware in recommendation tasks and so, it is inappropriate to provide movies to her/him in the morning even though such movies are the most relevant to her/him.

Given a training set which is a rating matrix and a user who requires recommendations, CF algorithm tries to predict rating value on items which are not rated by this user. After that CF algorithm makes a list of such items arranged in order of predictive rating values and recommends this user such list. In other words, CF algorithm constructs a predictive function *R2*  [1] whose cross domain is the Cartesian product of a set of users *U* and a set of items *I*. The domain *U x I* is also called rating matrix. Co-domain of function *R2* is a set of predictive ratings denoted *R*.

*: U x I → R*

Function *R2* so-called traditional 2-dimension (2D) mapping doesn’t consider contextual factors now and it can lack of information necessary to highly accurate prediction. Suppose contextual information including location, time and companion are added to prediction process, the 2D function *R2* becomes the 3-dimension (3D) mapping denoted as below:

*: U x I x C → R*

Note that *C*, *U x I* and *U x I x C* represent context domain, 2D (cross) domain and 3D (cross) domain, respectively. In order words, function *R3* gives recommendations to user under circumstances which specified as contextual information.

Although context has many different types, we can reduce these types into three main types in order to answer three question forms: when, where and who.

* Time type indicates the time when user requires recommendation, for example: date, day of week, month, and year.
* Location type indicates the place where user requires recommendation, for example: theater, coffee house.
* Companion type indicates the persons with whom user goes or stays when recommendation task is required, such as: alone, friends, girlfriend / boyfriend, family, co-workers.

Contextual information is organized in two forms: hierarchical structure [2] and multi-dimensional data (MD) model.

According to hierarchical form, the context domain *C* is defined as a set of contextual dimension ***K*** = (*K1, K2, K3,…, Kn*). **K** is represented in hierarchy structure, whose attributes *Ki* (s) are associated with the ascending order of fine level. For example, given attributes *Ki* and *Kj* where *i < j*, *Kj* is finer than *Ki* and so *Ki* contains *Kj*. It is easy to recognize that *K1* is the coarsest attribute which contains all remaining attributes *K2, K3,…, Kn*. Each *Ki* contains values at the same level *i* and it can be split into finer levels.

Figure 1.1 Contextual dimension example.

In above example, *K2* = {City, Province}, *K3* = (City → District, City → Suburb district, Province → District, Province → Suburb district)

According to MD form, context domain C is defined as the Cartesian product of *n* dimension, *C =* *D1 x D2 x… Dn.* Each dimension *Di*, in turn, is a set of attributes, *Di =* (*ai1, ai2,…, aik*). For example, suppose *C* has only one dimension of time denoted *D1 = Time* (*day of week*). So, the cross domain *U x I x C* of predictive function *R3* constitutes a 3D cube: *User* (*name*), *Item* (*book type*) and *Time* (*day of week*). Each block in this cube is assigned a rating which is the predictive outcome of function *R3*.

Figure 1.1 MD cube.

Above figure indicates that given user “John”, item “novel book” and time “Sunday”, the predictive outcome of *R3* gains the highest value 5. There are three approaches [1] to apply context into recommendation process:

* Contextual pre-filtering: Firstly, given context *c C* is used to select user-item pairs (*u, i*) which are more relevant to this context, leading to obtain the aware-context cross domain *U x I*. After that traditional 2D function *R2* is taken on such cross domain.
* Contextual post-filtering: Firstly, traditional 2D function *R2* is used to produce the list of recommended item. After that context *C* is used to fine-tune this list in order to remove irrelevant items according to concrete context.
* Contextual modeling: The 3D function *R3* is used directly on context-aware cross domain *U x I x C*.

The basic idea of contextual pre-filtering is to project the 3D domain *U x I x C* on 2D plane, based a concrete context *c C* so that such 3D domain is reduced to 2D domain *U x I*. Let ∏c be projection operation based on condition context *c*, we have:

*= U x I*

(*u, i, c*) *=* (*u, i*)

The concrete context *c C* which is strict projection condition can make the cross domain *U x I* small or sparse, causing low predictive accuracy. So generalization technique is used to make projection condition loose, namely the exact condition *c* is replaced by the more general condition *c’*. For example, the context *c* = ‘Saturday’ which indicates that “Mary prefers to go shopping on Saturday” is replaced by more flexible context *c* = “Weekend” because she can like going shopping on Sunday if she often goes on Saturday. The general context not only expends the space of potential recommendation solutions but also improve predictive accuracy.

The essence of contextual post-filtering is to fine-tune the raw recommendation results taken from predictive function *R2* which didn’t consider contextual factors before. Consequently, this method tries to figure out user’s context-aware interests, preferences or attributes by using some artificial intelligent and mining techniques and apply such attributes into raw results so as to remove out irrelevant items or change their ranks in final recommendation list. For example, given context *c* = ‘evening’ which indicates Mary’s interest “watching movies in the evening”, contextual post-filtering approach will remove all of news or sport events from her recommendation list.

The strong point of contextual pre-filtering and post-filtering approaches is the ability to take advantage of legacy recommendation algorithms not taking account into contextual factors.

The essence of contextual modeling is to incorporate directly contextual information into predictive function *R3* and so, *R3* is constructed as inference model such as data mining, machine learning, heuristic model or statistical model, etc. The 2D predictive function isn’t used in contextual modeling. It implies that the strong point of this method is pure and powerful inference mechanism. It opens a new trend in context-aware recommendation research, giving a lot of prospects although a few related techniques are extensions of 2D algorithms.

The approach in this paper is the hybrid of contextual post-filtering and contextual modeling. Hence a logistic inference model is used as the post filter to make recommendations more relevant to users according to contextual factors.

**2. Glance over logistic regression and mining frequent itemsets**

**3. Combine logistic regression with mining frequent itemsets**

**4. Evaluation**

**5. Conclusion**

**Reference**

1. [Ricci, Shapira, Kantor 2011]. Francesco Ricci, Lior Rokach, Bracha Shapira and Paul B. Kantor. Recommender Systems Handbook. Springer Science + Business Media, LLC 2011, ISBN: 978-0-387-85819-7.
2. [Palmisano, Tuzhilin and Gorgoglione 2008]. Cosimo Palmisano, Alexander Tuzhilin and Michele Gorgoglione. Using Context to Improve Predictive Modeling of Customers in Personalization Applications. IEEE Transactions on Knowledge and Data Engineering, vol. 20, no. 11, November 2008.
3. [Montgomery, Runger 2003]. Douglas C. Montgomery, George C. Runger. Applied Statistics and Probability for Engineers, 3rd. ISBN: 0-471-20454-4. John Wiley & Sons, Inc. 2003.