Context-Aware Personalization Using Neighborhood-based Context Similarity

Abayomi Moradeyo. Otebolaku\* and Maria Teresa Andrade\*#

\*Center for Telecommunications and Multimedia, INESC TEC, Faculty of Engineering, University of Porto, Portugal.

#Faculty of Engineering, University of Porto, Portugal

Corresponding Author: Abayomi Otebolaku

Email: abayomi.otebolaku@inescporto.pt

Phone: +351222094299

Abstract---With the explosive volume of online multimedia content, people with handheld devices seeking to access content of interest become more and more confused and disillusioned. Deciding on the most relevant content to consume among the deluge of available alternatives while going about their daily activities is increasingly difficult. Context-aware personalization with the capability to learn user’s contextual preferences has been proposed as an effective solution to address this problem. However, existing context-aware personalized systems rely on rating information explicitly obtained from users in the contexts of use. Therefore, these systems suffer from the so-called coldstart problem that arises when personalization systems are lack adequate knowledge to make suggestions for users, particularly the first-time users. This happens either because not all users especially in mobile environments that provide ratings every time they consume content or because such users are using the system for the first time. In this paper, we present an analysis and design of a context-aware personalized system capable of minimizing the coldstart problem in a mobile multimedia consumption scenario. Our proposed solution emphasizes the importance of similarity between contexts of consumption based on the traditional K-Nearest Neighbor algorithm and Pearson Correlation model to minimize coldstart problem. Validation of the proposed system via experiments and by comparing evaluation results of contextual and non-contextual personalization approaches show that the proposed system can effectively provide personalized content to mobile users.

Keywords— Context-awareness; personalization; neighborhood; mobile users; handheld devices; user profile.

## **1 Introduction**

The vast majority of handheld device users are increasingly becoming confused and disillusioned concerning decision on the appropriate and relevant multimedia content to consume as they go about their daily routines. On one hand, they are confused because of the unprecedented volume of available multimedia content they can now access online via the Internet and multimedia enabled handheld devices. On the other hand, mobile users are disillusioned because the content they decide to consume is often not relevant to their current contextual situations [1-5]. Therefore, the content delivered to them, most time, does not match their preferences. Imagine for example, a system suggesting latest movies to a user while she is seated at her desk working in the office! Additionally, in the course of searching for suitable content, users waste invaluable time and efforts with no satisfying consumption experience. Thus, to adopt handheld devices as the entertainment platforms for multimedia consumption, suitable technology is required to assist mobile users to obtain, either implicitly or explicitly, suitable multimedia content from myriad competing alternatives, matching their interests and contexts.

Personalized recommendation techniques and similar solutions providing this kind of functionality are common. For example, many massive online multimedia content providers have adopted and incorporated excellent recommendation services to provide their customers with effective and efficient suggestions either implicitly or explicitly [3]. However, most of these systems are designed to maximize business interests rather than user’s interests, using user provided item ratings [4]. These traditional approaches, as they are, cannot be used effectively in mobile environments because it is difficult, if not impossible, to obtain implicit user preferences directly from explicit ratings in such computing environments because preferences of mobile users are not fixed as their contexts change. The traditional approaches assume that user preferences are fixed and thus when user’s context changes, such system may not effectively provide relevant content matching the user’s new contexts. For example, a user may prefer to listen to music when walking or jogging in the neighborhood, whereas, the same user may prefer to watch romance movies while at home on rainy weekends. Similarly, because mobile users are prone to the so-called new user (c*oldstar*t) problem, traditional approaches that rely on explicit user ratings rather than on user’s contextual information cannot be effectively applied in mobile environments. This problem reflects the fact that users have not provided enough ratings in each context of consumption to allow adequate learning of their preferences [7].

To address the challenges related to applying traditional personalization techniques in mobile environments, context-aware personalized recommendations have been proposed, assisting mobile users to obtain content according to the user’s contextual preferences [1-2, 7, 9, 11-14, 17, 20].

Fig 1 illustrates a typical architecture of a context-aware personalized recommendation system, exemplifying the system proposed in this article. Thus, in this article, we discuss in detail question bothering on designing this type of system, using similarity between a new user’s contexts and those of other likeminded users to determine the new user’s contextual preferences.

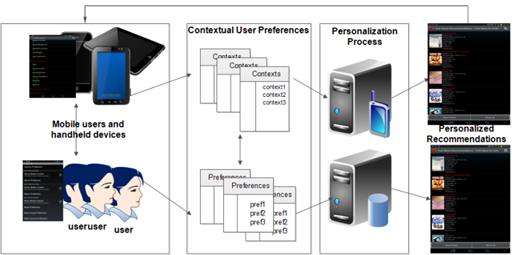


Fig 1 A simplified architecture of a contextual personalized recommendation system

The proposed solution uses contexts of the new user to identify likeminded users or so-called neighbors instead of using explicit item or user ratings, which are prone to the *coldstart* problem [7]. We show that by using context similarity, based on neighborhood model to determine the so-called neighbors of the new user, it is possible to build a simple and yet effective context-aware personalized service that can provide relevant multimedia items to first time users in mobile environments. First, such systems determine the new user’s current context, and second it uses the identified context information to discover users who have been in the same or similar contexts, including the content they consumed in those contexts. Third, and finally, it uses the preferences of these newly identified neighbors to predict the preferences of the new user. With this approach and based on our experimental validation we conclude that it is possible to minimize the *coldstart* problem.

The rest of the paper is organized as follows. Section 2 presents an overview of related work. In section 3, detailed description of the proposed system is presented. In section 4, the paper describes experimental evaluations of the system and discusses the results obtained. Finally, section 5 draws some conclusions and provides the research’s future direction.

## **2 Background and Related Work**

Conventional personalized recommendation systems have been designed to deliver multimedia content that matches user’s preferences timely and effectively [2]. However, these systems focus chiefly on suggesting relevant items from millions of alternatives to users on the assumption that user’s tastes and preferences are fixed [5, 12]. They assist target users to discover relevant items explicitly using three popular methods: (1) evaluations of previously consumed multimedia content given by the target user, a process popularly known as content based filtering (CBF), (2) using evaluations of items consumed by likeminded users, otherwise known as collaborative filtering (CF) [8] and (3) using a combination of these approaches in what is popularly called hybrid recommendations [1-3]. Although these conventional personalized recommendation techniques have been highly successful, however, they are usually designed to address explicit recommendations of multimedia items. They anticipate explicit requests from users before providing recommendations, using rating information provided by the users or information about the content consumed by the users. Thus, because there is no guarantee that users will always provide rating information for multimedia content they consume, both CF and CBF suffer from the coldstart or new user problem [4, 7]. Due to coldstart, these systems are not reliable to provide effective recommendations for new users, i.e. those users that have never been provided with recommendations by the system. This problem is usually addressed by asking these new users to provide their preferences explicitly in order to provide recommendations [7]. Additionally, unlike collaborative based recommendation systems, content-based recommendation systems suffer from overspecialization, which occurs when they provide users with only those items that are similar to those they have consumed in the past [1-3,5]. Furthermore, if a user has no record of consumption, it becomes a problem to provide such users with recommendations. To address these issues, hybrid recommendation solutions attempt to mitigate these problems while trying to avoid the weaknesses of both content-based and collaborative-based recommendations but, profiting from their strengths. Nonetheless, conventional hybrid recommendation proposals cannot be effectively applied in mobile environments because they are designed based on the assumption that mobile user’s preferences are fixed irrespective of their contextual situations and thus, they still suffer from the *coldstart* problems [3-5].

In the last years, contextual recommendations have been proposed to address these weaknesses by taking into account the contextual situations in which users consume content [1-2]. These systems aim to use context-awareness to learn the preferences of users in different situations and then use this knowledge to provide personalized recommendations. The definition and information on how context information can be incorporated into recommendation systems have been presented by Adomavicius et al. in [2]. Meanwhile, even though existing work proposed provide methods for using contextual information to provide relevant information to users, these contextual approaches rely on explicit ratings provided by users in specific contextual situations where they consume multimedia content [3]. But in mobile environments, users usually do not provide ratings when they consume content [3, 5]. Additionally, asking users to provide ratings every time their context changes can only easily become boring and annoying. Thus, *coldstart* problem remains an issue.

Zhu et al. [10] proposed a generic context-aware recommendation system that takes as input user preferences, user contexts and device capabilities to provide multimedia content recommendations to smartphone users. The proposed system uses content-based recommendation method, using the Vector Space Model (VSM) to learn user’s preferences. The system uses a Bayesian classifier to evaluate contextual information against media items consumed by users in those contexts. This system relies on the user’s contextual consumption to suggest new items. It means that for a new user, it would be difficult to provide recommendations.

Chen [12] proposed a context-aware collaborative recommendation system that predicts user’s preferences in different contexts. It extends traditional neighborhood based collaborative recommendation system, using Pearson correlations to provide recommendations to users based on explicit ratings given by likeminded users in similar contexts.

Another interesting work uses daily activity contexts and other contextual information of users to provide personalized content according to the current activity the user performs using collaborative recommendation [11].

Also, a recent proposal by Alhamid et al. [14] uses social tags and user rating information to personalize multimedia content search in a context-aware collaborative recommendation system. Nevertheless, using rating information would lead to poor recommendation quality because in mobile environments, as earlier iterated, users usually do not always give ratings of content they have consumed. Additionally, it provides recommendations only if users make explicit requests.

In our previous work, we described how contextual information can be incorporated into a content-based traditional recommendation system via a novel user profiling approach to provide multimedia content recommendations [9]. However, in that project, we only used content consumed in the past by the users to provide contextual recommendations personalized for the current context. Thus, the coldstart problem was not our focused problem but demonstration of how context information can be used to provide relevant recommendation by assuming that the system has adequate past consumption information of the target users.

The systems discussed so far, even though they incorporate user’s contextual information, they were designed to request explicit preference information from users in contexts where they consume multimedia content. Therefore, despite these excellent proposals, the *coldstart* problem has not been addressed adequately.

In this paper we show how similarity between contexts of new users and those of other(existing) users can be used to minimize the *coldstart* problem without the requirement of explicit rating information from users as proposed in previous solutions [10-12, 14].

## **3 Context-Aware Personalization: Our Proposed Approach**

The proposed system was designed based on the traditional nearest neighbor based collaborative recommendations using Pearson Correlation Coefficient (PCC) algorithm to predict the target user’s preference in the current context, according to the consumption experiences of his contextual neighbors as illustrated in Fig 2. The contextual user’s consumptions have been modelled using contextual profiles, which consist of contextual preferences of each user, representing their consumptions in specific contexts. The preferences of each user is thus defined as follows:

(1)

(2)

(3)

(4)

Where, is user k (k = 1, 2, 3, etc.) and represents his preference for multimedia content in context . We define a new user, whose preference  for multimedia for item  in the context  we want to predict.

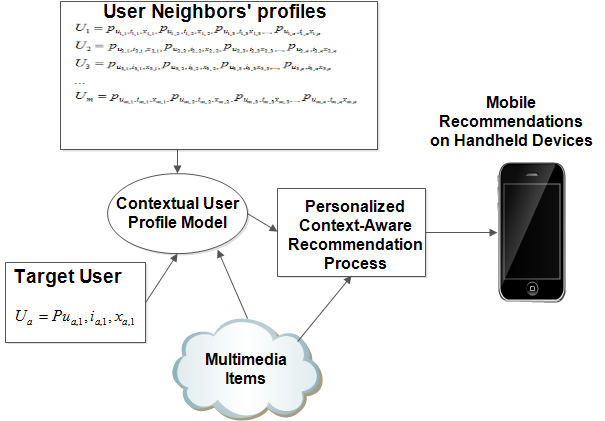


Fig 2 Contextual user profiling and personalized recommendations

### 3.1 Learning Contextual User Profile

Considering the definition in the last section, the contextual profile usually consists of information concerning the user’s consumption history, which is obtained either explicitly or implicitly as described in our previous article [9]. In that article we described how the contextual user profile model can be used for context-aware content-based recommendations. But here, we present an overview of the contextual user profile model and explain in the next section how it was used in conjunction with the similarity between users’ contexts to determine their preferences in a neighborhood(collaborative) process. As one of the key requirements to personalize multimedia content for mobile users, a contextual user profile learns the user’s contextual preferences by representing the multimedia content a user would prefer to consume taking into consideration her contextual situations. In the user profile model, as illustrated in Fig 3, the contextual situation represents information that characterizes the mobile user’s consumption, such as her activity, location, weather information, location illumination or noise level, time, etc. Let *U* = *{ u1,u2,u3,..un}* be a set of *N* mobile users. Each user *uk* is defined by a set of preferences defined by equation *(4).* Each preference is defined by a set of category, *Ca = {ca1,ca2,ca3,…,can}* and each category is characterized by a set of genres, *Gr = {gr1, gr2, gr3,…, grn}. F*or example, movies are characterized by genres such as action, drama, comedy, family, etc. It also includes a set of properties characterizing the content consumed by the user at lower levels than the category and genre. These properties, defined by set A = {a1, a2, a3,…, an}, may include elements such as language, duration or year of publication of the content, etc.The category set represents the classifications of the consumed contents, e.g. movie, news, music, etc.Additionally, we define a set of high-level contextual information, *C = {c1, c2, c3, cn}* associated with each preference of the user uk, . In this definition, context *C* has a relatively complex structure because many contexts could characterize the consumption of a user, reflecting the complexity of representing contextual profile concisely.



Fig 3 Contextual user profile model

The user preference model is thus defined in a general form as follows.

*Y = f(x1, x2,x3,…,xn)* (5)

Where x*1, x2, x3,…, xn* are the user attributes and the preference information, which are characterized by set *A, Gr and Ca*. *Y* is the dependent variable to be determined, and function *f* is the predictive function learned via some classification algorithm. For example, the system could predict the media item that user *uk*would like to consume using this model. However, the model defined by (5) does not account for contextual preferences of users but only defines the traditional user preferences. To incorporate contextual information into the model, we modify the above definition as follows.

Y = *f*(*x1 c1,x2c2,x3c3,…,xncn*) or Y = *fcn*(x*1,x2,x3,…,xn*) (6)

Where c*i*,…,c*n* represent the contextual information and *fcn* is the function for identifying users’ contextual preferences. This model can be used to learn the user profile in two modes. First, it works in the contextual mode and, second, in the non-contextual or traditional mode to determine the user preferences with and without using contextual information respectively. The traditional mode (non-contextual mode) is executed in situations where user’s contextual information is not available or when it is difficult or impossible to acquire. In this mode, an entire profile considering user consumption history (if any) is used. Alternatively, it could use the consumption history of users who are similar to the current user to learn the user preference. The default mode is the contextual mode in which user’s contextual information is used to learn the user’s preferences. In the evaluation in section 4, recommendation Average Precisions (AP) based on these two models are compared.

To learn user’s multimedia content consumption preferences based on the above profile model, the proposed system adopts the relevance feedback method. Relevance feedback is a way to obtain a user’s opinion on recommendations; it can speed up user preference learning process and improve the quality of recommendations [16]. There are two methods in the literature used for learning user preferences based on user feedback, namely implicit and explicit relevance feedback [16]. In implicit user feedback, on one hand, the system observes the content consumptions by the user and records information about the consumed content. In our case, such information includes category, genre, property, etc. of the multimedia content and most importantly the context in which users consume this content. Our implicit user feedback process, without asking for any information from the user, assigns a relevance value to the consumed content using the content metadata information as well as the context of consumption. On the other hand, the explicit user feedback involves asking user to provide ratings or some form of evaluation of the relevance, such as, like or dislike of the recommended items. In mobile environments, this approach is obtrusive as it distracts and consequently bores users, discouraging them from using recommendation system, which consistently requires them to provide rating information every time they consume content [16]. For this reason, we adopted an implicit user feedback method. Our approach neither ask users for explicit rating information nor measure how long a user has spent during the consumption of a particular content, for example. Measuring time spent on recommended items as rightly opined by Bjelica [16] may lead to wrong conclusions. Thus, rather than measuring time spent when consuming content, we used a combination of contextual user profile learning model and the context in which users respond to recommendation either by clicking or by not clicking (the device’s screen, for example) to learn the user’s feedback on the recommended items. It then extracts implicit information about such items to learn whether users like or dislike such items by assigning what we call relevance values to the extracted information, such as category of the content, genre of the content, etc. taking into consideration the contextual situation of the user at that given time. The system automatically assigns these relevance values in two numeric formats as illustrated in (7) and (8) consisting of *weight* *w* and *lifetime* γ parameters.

|  |  |  |  |
| --- | --- | --- | --- |
|  | (7) |  | (8) |

The weight *wi* provides information on the number of times the user has consumed items of that *category-genre-property* in that specific context. The lifetime parameter provides an indication of the time elapsed since the last consumption. Its value is set to 1 when the user consumes a content of that category-genre-property, and periodically decrements it if the subsequently consumed content does not belong to such category-genre-property. Smaller values indicate that the user has lost interest in that type of content, regardless of the value of its weight or contexts. In practice, it allows to give more importance to items consumed recently and less to those consumed very long time ago. The factor α has a value in the interval [0:1], assigning less or more importance to new category-genre-property. For newly created category-genre-property, α assumes the value 1; otherwise its value, determined based on experiments, is 0.5. The factor β is the score given by the system and can assume the values 1 or -1. The value 1 indicates that the user has consumed a new item having the characteristics as described by that category-genre-property (i.e., indicates a preference of the user). The value -1 was used to indicate that the user has rejected an item with those properties.

The *lifetime* parameter as explained ealier is set to 1 for matching items in the user profile, by assigning the value 0 to the factor *t*. The factor, *t,* represents the number of days elapsed since the last time the user has consumed an item with the characteristics described by the profile. With equation (8), the relative importance of the category-genre-property of an item consumed by a user remains above 0.9 in the first 30 days after it has been visited, rapidly decreasing to zero after that period (non-negative values are automatically converted to zero) [9]. For all other *category-genre-property* in the user profile, the update of the *lifetime* parameter is performed by linearly increasing the value of *t*. This way category-genre-property of a multimedia content that has been consumed before, but has not been seen or consumed for a long period will have either low or no impact on the user preferences evaluation.

### An high-level overview of predicting new user’s preference in the current context using similarity between new user’s context and contexts of likeminded users

The user profile model presented in the last section assumes that the system has adequate knowledge of those users represented by the model. Thus, for a new user, the model will provide poor recommendations.

To address this problem, we used contextual information and preferences of existing users as defined by the contextual user profile model to learn the preferences of a new user considering his present context.

To realize this process, our system first determines the new user’s current contexts. This is possible because we have developed a dynamic context recognition model as part of our broader personalization system [15, 18]. First, the model acquires low-level context data from the user’s handheld device to classify and identify their contextual situations. The detailed description of how this was realized is out of the scope of the present article [15]. Second, having identified the current context of the new user, the system then uses this context to search the profile of existing users to find those users with contexts that are similar to the new user’s context. Those users with similar contexts as the current context of the new user are considered as his neighbors or friends. Third, by identifying these neighbors, the *n top* most similar users are filtered, based on the order of the magnitude of their contextual relevance to the new user. The profile of the user with the highest relevance value, as well as those of the other *n-1* users, is used to compute the preference prediction for the new user. Fig 4 illustrates the work flow of the context-aware collaborative personalization. The process for identifying such relevant contexts and for using it to predict the new user’s preference is described in the next section.



Fig 4 The Proposed Context-Aware Collaborative Personalization Process Workflow.

### 3.3 Finding users with similar context to the new user’s context

The rationale for finding users whose contexts are similar to the new user’s context is to determine the content that would likely interest the new user in the current contexts based on the contents consumed by users with similar contexts. We designed our solution based on the hypothesis that users with similar contexts usually share similar interests. Therefore, if we can establish the contextual situation of a new user, this information can be used to identify users who prefer certain kinds of contents in similar contexts. For example, when people gather at a cinema, it is most likely that they prefer movies of similar genres and properties and thus, movies or other similar content can be suggested. Thus, the similarity of contexts between a new user and existing users can be used to compute the preference of the new user in such context.

We deduced from the illustration above that context information could come in various types. For example, location type, time, activity type, etc. For each context type, we define a similarity function as used in [12]: *simt (cn,ct).* Where *cn* is the new user current context and *ct* is the context of other users. In order to determine the similarity between a context and another, we defined two classes of contexts. First, the categorical contexts that can assume unordered values, e.g. activity contexts are examples with values such as *Walking, Running, Jogging*, etc. Second, the numeric contexts, which consist of ordered values, examples of such contexts *are temperature, noise level, illumination*, etc.

To obtain the similarity between two numeric contextual preference cases, we used (9).

(9)

Where , Max is the maximum numerical value of the *context* and Min is its minimum value. For example, let us take the noise level as an example, if Min is 10 dB, Max = 150dB, current, is 50, and = 60 then the value of p is 0.0714. In this case, the similarity between and is 0.93.

To obtain similarity between two categorical features, a partial matching scheme similar to Lee & Lee [17] was adopted, using the domain knowledge. In this case, predetermined weighted scores are assigned to each categorical context. Tables 1, 2 and 3 illustrate examples of such weighted scores used as similarity values, which are automatically assigned by the system for matching activity, time and location context information. Contexts that match exactly are assigned 1.0, and those not matching, are maybe assigned less value or zero.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table 1 Sample Weighted Scores for User Activity Contexts | | | | | | |
| New user  Old user | W | R | J | St | S | L |
| Walking (W) | 1.0 | 0.2 | 0.3 | 0.1 | 0.1 | 0.1 |
| Running (R) | 0.2 | 1.0 | 0.5 | 0.2 | 0.3 | 0.1 |
| Jogging (J) | 0.3 | 0.5 | 1.0 | 0.1 | 0.3 | 0.0 |
| Standing (St) | 0.1 | 0.2 | 0.0 | 1.0 | 0.3 | 0.0 |
| Sitting (S) | 0.1 | 0.1 | 0.0 | 0.3 | 1.0 | 0.3 |
| Lying (L) | 0.0 | 0.0 | 0.0 | 0.0 | 0.2 | 1.0 |

|  |  |  |  |
| --- | --- | --- | --- |
| Table 2 Sample Weighted Scores for Day of the Week Context Information | | | |
| New user  Old user | Weekday | Weekend | Holidays |
| Weekdays | 1.0 | 0.1 | 0.2 |
| Weekend | 0.1 | 1.0 | 0.9 |
| Holidays | 0.1 | 0.9 | 1.0 |
| … | … | … | … |

|  |  |  |  |
| --- | --- | --- | --- |
| Table 3 Sample Weighted Scores for Location Information | | | |
| New user  Old user | Cinema | Office | Home |
| Cinema | 1.0 | 0.0 | 0.0 |
| Office | 0.0 | 1.0 | 0.0 |
| Home | 0.0 | 0.0 | 1.0 |
| … | … | … | … |

3.3 Predicting new user preference

Last section describes how to determine the similarity between a new user’s current context and contexts of existing users. For all users with similar contexts, the system obtains the similarity values computed using the strategies described in the last section in descending order of magnitude and selects the first n users. Among these n users, consumption preferences of the user with the highest similarity, who we call a reference user and those of other *n-1* users are then filtered to predict the preferences of the target (new) user. In this section, we explain these steps involved to realize process.

The first step is to measure the correlation between the each neighbor, designated as user uk. We use the popular traditional Pearson coefficient to measure the correlation between consumption preferences of the neighbor user uk and each other neighbor vk [8, 12]. We designate his content preference in context ct for content *i,* having *category-genre-property* as similar and that of the other neighbor as .We use (10), which combines the preferences of neighbor uk and every other neighbor vk into a weighted average, using the correlations of these neighbors as the weights.

(10)

*Where*  and are the averages of weight w defined in equation (7) for uk andvk respectively. returns the relevance of consumption preferences of both users in context ct for item *in*, containing category-genre-property.

We defined a weighted preference ( of the reference user uk for an item *i,* which he consumed in contexts similar to the current context *cn* of the new user of the new user using context similarity as weights. Because the context is multidimensional e.g. each location of the user is assumed to be characterized by e.g. the activity the user engages in that location, the location illumination, the time of the day at the location, the day of the week, etc. the similarity of these context dimensions is computed and summed as illustrated in equation (11). k in (11) is a normalizing factor such that the absolute values of the context similarity sum to unity.

(11)

To obtain the preference for the new user, we use equation (12).

(12)Where n is the number of contextual neighbors of the new user and k is a normalizing factor.

|  |  |  |
| --- | --- | --- |
| Table 4 Example User Profile Information | | |
| Level | C-G-P | Possible values |
| Category | Category | Movies (others are news and music) |
| Genre | Genre | action, animation, comedy, drama, documentary, epic, horror, politics, sci-fi, sports, thriller, etc. |
| Property/  media item’s context | Language | English, French, German, Italian, Portuguese, Spanish, other |
| Country | Canada, France, Germany, Italy, Portugal, Brazil, Spain, UK, USA, etc. |
| Date | old (>5 years), recent (<5 years, >1 years), present (<1 year) |
| Duration | short (<30 mn), medium (>30 mn, <75 mn), long (>75 mn) |
| user’s context | Context | @homeWeekDayMorningSitting,@homeWeekDayAfternoonLaying, @homeWeekDayEveningStanding,@homeWeekendMorningSitting,@homeWeekendAfternoonSitting, @homeWeekendEveningWalking,@officeMorningSitting, @cinemaAfternoonSitting,cinemaMorningStanding, cinemaWeekendAfternoonSitting, cinemaWeekendEveningStanding,etc. |

## **4. Performance Evaluation**

In order to validate the proposed system, its experimental evaluation was conducted. We evaluated the ability of the system to provide relevant content to mobile users who have inadequate information i.e. new users, using an Android based mobile application platform we have developed for providing contextual recommendations for mobile users [18]. We also evaluated the impact of neighborhood size on the quality of recommendations and recommendation time. Before discussing in detail the evaluations, next we provide information on the experimental data used as well as experimental setup.

### 4.1 Experimental Data

In the absence of suitable large-scale context-aware personalization data for our proposed system, we obtained two sets of data for the experimental evaluation. The first set of data consists of over 4500 movie metadata obtained from the Movie Database (themoviedb.org), further enhanced with additional metadata retrieved from the Internet Movie Database (IMDB). This metadata set contains 23 separate movie genres, to form over 4500 multimedia content records. Each record contains on the average, 3 different genre labels: language, cast, country, duration, and release date as properties characterize those genres.

The second set of data consists of contextual user data obtained from 200 volunteer users. The data represent category-genre-property of content, especially movies, consumed by the users in different contexts. About 80% of these volunteers are students, 11% are researchers, 2% are professors, 9% are professionals, and 2% are others. For experimental purposes, we only collected data on the genre of movies (category) consumed the language, the country, etc. (as properties). At the same time, the contextual information of the consumed content was also collected. Table 4 illustrates examples of context data associated with the collected user data. Note that in Table 4, C-G-P represents category-genre-property of the content. These sets of data can be instantiated into the context user profile profiles implemented in the server side of our application, thus this information served as the contextual consumption history of mobile users.

### 4.2 Experimental Setup

For evaluation purposes, we have implemented the proposed solution as part of our existing application: a mobile client application running on handheld devices and a server application, both deployed in an experimental setup as illustrated in Fig 5. The server application implements processes, including the contextual user preference model. We used these applications to provide contextual recommendations of content to mobile users as illustrated in Fig 6 and Fig 7. The data described in the last section were instantiated in the server as contextual user profiles. Then for each user profile and based on the contextual preferences they contain, recommendations were generated at 10 different times. The personalized multimedia content application receives recommendation requests from the mobile client, including the user’s contextual information, it then processes the requests to determine likely preferences of the user in that context.

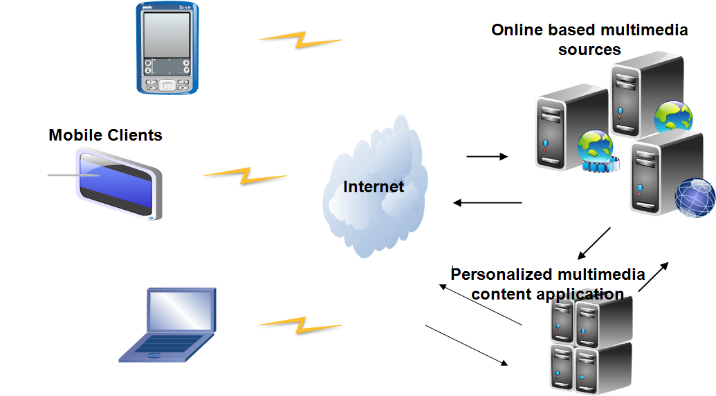


Fig 5 Experimental evaluation setting

With this knowledge, it determines the category-genre-property corresponding to those preferences, and then retrieves the set of content with similar genre and properties from online-based sources, ranking them according to the preferences of the user. The top k items in the set are then provided as recommendations and displayed on the device screen. Users can then click to play preferred contents

.

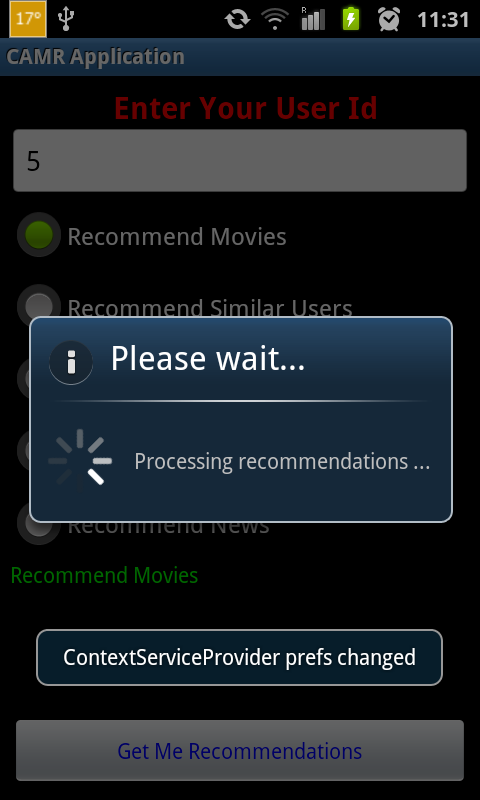


Fig 6 Recommendation Process of the experimental Application

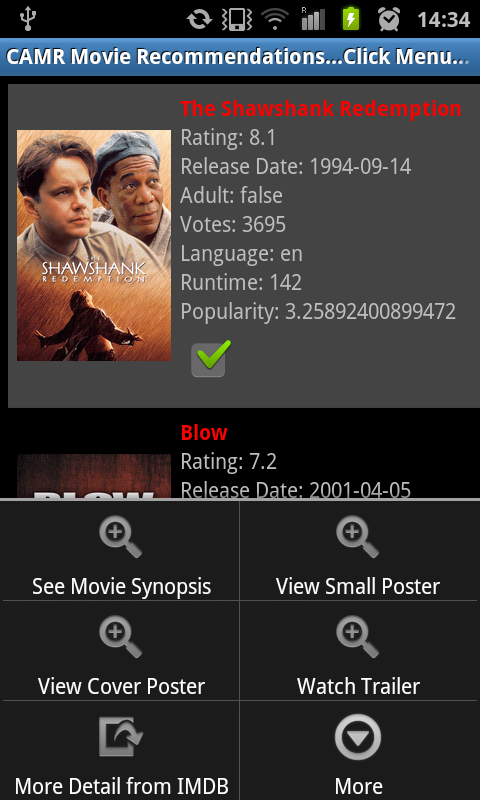


Fig 7 Recommended Items provided to users for evaluation

As evaluation metrics, we decided to compute average precisions (AP@k) and mean average precisions (MAP@k) at top k recommendations as measures of the accuracy of the personalized content generated for each recommendation and for all recommendations respectively [20].

Average Precision (AP@K) is the mean precision score obtained for each relevant item at top k recommendations in the test cases for every test user, computed as follows.

(13)

Where *m* is the number of relevant items and = 0 if item *i* in the set is not relevant (not selected by the user as relevant). For these test users, recommendations were generated 10 times, with k = 5. This means each recommendation consists of a set of 5 items. Each time recommendation is generated for each test user, the average recommendation precision (AP@10) is computed for him/her.

#### 4.2.1 Evaluating new user problems

In the series of experiments conducted, the performance of the proposed system is evaluated for 20 test users. These 20 users are those with little or no information in the user profile model some of who we consider as new users. The validation was carried out in four carefully selected contextual situations, which we designate as context types as follows:

*(A) @homeWeekendNightSitting, (B) @cinemaWeekendNightStanding,*

*(C) @homeWeekDayNightStanding, (D) @cinemaWeekDayNightSitting*

*(E) @noContext.*

We also executed the experiments without using context information (No-Context) to compare performances of the model in those situations. We chose these contexts since it is not possible to evaluate the system under every possible context. Thus, it is obvious from the above context types that each consists of four contextual components. First, we have the location, second, third and fourth are the day of the week, time of the day and activity contexts respectively. Fig 8 shows the average precision of recommendation under each context type, and in no-context situations. The no-context situation is regarded as the baseline or the traditional personalized recommendation process. The most important conclusion from this figure is that in those four contextual situations (context types), the system’s recommendation AP is higher than in no-context situations.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table 5 Mean Average Precision and Confidence Level(Confidence Interval of @ 95%) of AP@10 | | | | | |
| Context Type | A | B | C | D | E |
| MAP@10 | 0.640 | 0.559 | 0.474 | 0.412 | 0.224 |
| CL | ±0.0287 | ±0.0155 | ±0.0318 | ±0.0308 | ±0.0446 |

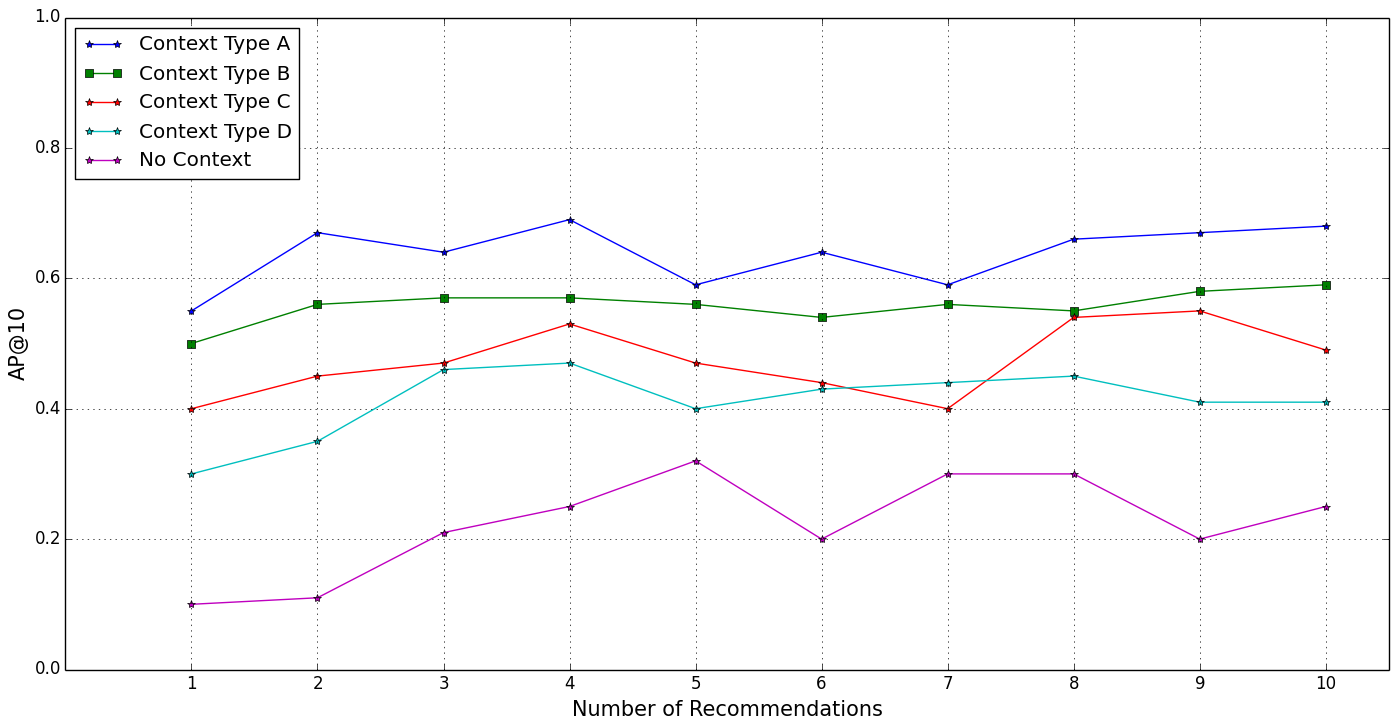


Fig 8 AP@10 of recommendations in selected contextual situations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table 5 Mean Average Precision and Confidence Level(Confidence Interval of @ 95%) of AP@10 | | | | | |
| Context Type | A | B | C | D | E |
| MAP@10 | 0.640 | 0.559 | 0.474 | 0.412 | 0.224 |
| CL | ±0.0287 | ±0.0155 | ±0.0318 | ±0.0308 | ±0.0446 |

In addition, when we look at the number of recommendations, the accuracies of the context based recommendations show clearly that even at smaller number of recommendations, the system is able to provide better recommendations than in no context recommendation. The low recommendation accuracies of the baseline evaluation underline the coldstart problem, where the system could not find adequate information to learn the new user’s preferences.

The result further shows that the system can provide recommendations for users with little or no information in the system as long as such new user’s context can be obtained and used in the recommendation process.

Furthermore, analyzing the AP under specific context types, *@homeWeekendNightSitting (A)* and *@cinemaWeekendNightStanding (B)* tend to provide better recommendations, although the recommendations in all context types performed better than the non-context condition. The reason is that naturally, and based on the contexts in which users of the system have consumed movie items in our experiments, the results suggest that more neighbors consumed movies in these two context types than in other context types. Thus, both produced slightly better performance than others did as can be deduced from the figure, further elaborated in Table 5 shows the statistical confidence level (CL) of the MAP@10 for each recommendation provided. The proposed solution is based on collaborative recommendation technique, where contexts of new users are utilized to identify those that could be called their friends or other likeminded users. Thus, it can be seen that generally, the quality of contextual recommendations under those context types tends to improve with increasing number of recommendations.

This is a typical characteristic of collaborative recommendations, because the more interactions users have with the system, the more information it has to learn the preferences of the users, and the better it is able to provide relevant recommendations. This explains why the recommendation’s AP improved as the number of recommendations increased.

#### 4.2.2 Impact of Neighborhood Size on the Quality of Recommendations

Although results of the previous experiments clearly show the advantage of using contextual information to minimize coldstart problem, nevertheless collaborative recommendation based on neighborhood methods require computation that grow with both the size of the user and items [19], we decided to understand the influence of the number of neighbors on the recommendation AP and the profile processing time. Thus, another series of experiments were conducted, where we varied the values of the neighborhood size. Fig 9 shows the relationship between the neighborhood size and the corresponding recommendation AP for each context type used in the previous experiments. We observe that the number of users whose contexts are used to predict the preferences of a user has significant impact on the average precision of recommendations. However, the AP responds differently to different context types as the number of neighbors increases. The AP steadily increases as the number of neighbors increases for all the context types including the No-Context. Thus, considering this sensitivity of the AP of the recommendations for the neighborhood size, we consider neighborhood size between 10 and 20 as optimal, with average precision reaching up to 70%. Fig. 7 shows the neighborhood size and the profile processing time. Note that the time increases with the size of neighborhood. For example, between the neighborhood size of 10, 15 and 20 are 105, 180, and 310 (ms) respectively. The significance of this result is that the recommendation quality improves with increasing number of neighborhood size. Nevertheless, when compared with the results in Fig 10, the profile processing time becomes a bottleneck because the recommendation time increased as we increased the neighborhood size.

Fig 9 Impact of neighborhood size on the AP@10 of recommendations for selected contextual situations

Fig 10 Impact of neighborhood size on the AP@10 of recommendations for selected contextual situations

#### 4.2.3 User based Evaluation

In addition to evaluating the overall recommendation quality of the proposed solution, case study scenarios were set up in specific contextual situations in controlled experiments. The aim of the user-centered evaluation is to provide a generalized performance of the proposed solution involving real users. We evaluated user’s satisfaction of the system to ascertain its usefulness and effectiveness to provide understanding of how it might perform with actual users.

In the experiment, another 20 individuals who were representatives of the system’s target users constituted the test panel, whereas 200 user profiles constituted the active contextual profiles in the server subsystem. Considering the difficulty of convincing people to participate voluntarily in this evaluation, we were not able to get all the 200 users whose contextual profiles were incorporated into the system to give feedback. However, judging by the number of users who participated in similar experiments and related work, we have a relatively higher number of participants. For example, evaluation of the system presented by Pessemier et al. in [11], had 16 participants. In [21] and [22], Oku et al. evaluated their systems with 5 and 8 participants respectively. Similarly, in [10] and [23] Yu et al. and Wang et al. evaluated similar systems based on 9 test users with 200 movie records and 10 test users respectively.

In the user evaluation process, we evaluated two contextual scenarios as our experimental case studies. The first case study is the home scenario where the user usually performs some activity characterized by other important contexts. These contexts and their values are as follows*. {Location = Home, Activity = Sitting, Day = Friday, Time = Afternoon, Illumination = Bright, Noise level = Normal}*  It then provides recommendations to users, based on this contextual information. The second case study is the office scenario where the system has to detect the user’s activity and other contexts and use this information to provide recommendations for the users. The detected contexts are *{Location = Office, Activity = Sitting, Day = Monday, Time = Afternoon, Illumination = Bright, Noise level = Normal}.* In both scenarios, the users then provide a satisfaction score for each recommendation. For example, in each scenario, 10 recommendations were provided in decreasing order of relevance to the user’s contextual situation, using the case study context-aware recommendation application.

We asked the test users to evaluate the degree of satisfaction of the recommendations on a 5-point scale, where “5”, the maximum point, means *very satisfied* and “1”, the minimum point, means *not satisfied*. We analyzed the feedback provided by these users. We converted the satisfaction scores into binary values as follows. All scores between 3 and 5 were considered satisfactory; those below 3 were considered not satisfactory. We collected all the scores and then calculated the average to obtain the overall satisfaction from the user’s perspective. The overall satisfaction obtained was 69%. Additionally, the final feedbacks from the test users were analyzed and were used to compute average precisions for each test user based on top 10 recommendations. On one hand, the mean average precision (MAP) obtained was 0.65, which shows that the proposed system can provide recommendations that are relevant to each user in specific context with confidence level of 0.65 ± 0.059 (95% confidence interval), thereby minimizing the number of false positives.

## **5 Conclusions**

This article presents the design of a context-aware personalization system that uses neighborhood-based approach and contextual information to minimize the influence of the coldstart problem, which is common in conventional personalization systems. The proposed solution focuses on implicit delivery of personalized mobile content to mobile users who use handheld devices as entertainment platforms. These devices come with the capability to sense user contexts via their embedded sensors and with multimedia enabled capabilities.

The main contribution of the current work is the usage of contextual information to identify other likeminded users or neighbors whose preferences or those similar would interest a new user i.e. addressing coldstart problem of a new user. We argue that in similar contextual situations, users tend to express similar preferences for media consumptions. Essentially, after identifying these users, their contextual preferences are then used to predict preferences of the new user. The contextual information of the new user is determined dynamically using our existing context recognition system [15].

Experiments conducted to validate the proposed system showed that using similarity between contextual information, the coldstart problem can be minimized without relying on rating information. Evaluation under carefully selected contextual situations showed consistently improved recommendation precision. Whereas evaluation of recommendations under no context condition showed poor performance, reflecting the coldstart problem. However, when we compared the results of recommendations under contextual situations with those under no context, the contextual recommendations showed clearly, how context can be explored to address the coldstart problem, which is obvious in the results as contextual recommendations produced better precision of up to 70% compared to that of no context that produced poor precision between 10% and 43%. Additionally, *Context Type A and* B showed the best performance*.* The better AP of contexts A and B compared with other context types as illustrated in Fig.8 for contextual recommendations is because the contextual situations such as *HomeWeekendNightSitting* and *CinemaWeekendNightStanding (i.e. Context Type A and* B)represent common contexts that most users in the user data prefer to consume movie contents.

In the user evaluation process using two contextual case studies, the results obtained also confirmed that contextual information plays a key role in improving the relevance of content suggestion without necessary collecting rating information from user every time they consumed content.

Further experimental tests on the impact of the neighborhood size on the recommendation AP showed improved AP with increasing number of neighbors. However, in another series of tests to evaluate the impact of the neighborhood size on the user profile processing time, we observed that the larger the size of the neighbors, the longer time it takes to process the user profiles. Therefore, there is a tradeoff between AP of the recommendations and processing time. Higher AP due to larger neighbor size means longer processing time. However, the AP of the system between neighborhood sizes of 5 and 15 we consider as optimal, considering the processing time and, although depending on the context type.

In the future, we plan to investigate alternative methods to improve the AP of the system and processing time, which tend to deteriorate with increasing number of neighbors. We would like to investigate some other advanced and robust collaborative approaches, such as combining neighborhood model and latent factor model in a context-aware collaborative recommendation process. The latent factor approaches such as Singular Value Decomposition (SVD) comprises alternative methods for collaborative filtering with better ability to uncover latent features for better explaining user’s observed preferences [13].

**Acknowledgments** This work is partly financed by the Portuguese Foundation for Science and Technology (FCT) with grant SFRH / BD / 69517 / 2010.

**References**

[1] Adomavicius, G., Tuzhilin, A., Sankaranarayanan R. & Sen, S. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. ACM Trans. Information Systems, vol. 23, no. 1, pp. 103-145.

[2] Adomavicius, G., & Tuzhilin, A. (2005). Towards the next generation of Recommender Systems: A survey of the State-of-the-art and Possible Extensions. IEEE Trans on Knowledge and Data Eng., 17 (6), 734-749.

[3] Xia, F., Asabere, N.Y., Ahmed A.M., Li, J. & X. Kong X. (2013). Mobile multimedia recommendation in smart communities: A survey. IEEE Access, vol.1, no., pp.606-624.

[4] Martin, F. J., Donaldson, J., Ashenfelter, A., Torrens, M., & Hangartner, R. (2011). The big promise of recommender systems. AI Mag. 32(3), 19–27, 2011.

[5] Yujie, Z., & Licai, W. (2010). Some challenges for context-aware recommender systems. In Proceedings of Workshop on Recommender systems at the 5th International Conference on Computer Science and Education, IEEE, pages 362—365.

[6] Kwon, H. J. & Hong, K. S. (2011). Personalized smart TV program recommender based on collaborative filtering and a novel similarity Method. IEEE Trans. Consumer Electron. vol. 57, no. 3, pp. 1416-1423.

[7] Krstic, M. & Bjelica, M. (2012). Context-aware personalized program guide based on neural network. IEEE Transactions on Consumer Electronics vol.58, no.4, pp.1301-1306.

[8] Desrosiers C. & Karypis, G. (2011). A comprehensive survey of neighborhood-based recommendation methods. In: Recommender Systems Handbook, pp. 107–144.

[9] Otebolaku, A.M. & Andrade, M.T. (2014). Context-aware user profiling and multimedia content classification for smart devices. 28th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp.560-565.

[10] Yu, Z., Zhou, X., Zhang, D., Chin, C.Y, Wang X. & J. Men, J. (2005). Supporting context-aware media recommendations for smartphones. IEEE Pervasive Computing, Vol.5, No.3, pp. 68-75.

[11] Pessemier, T.D., Dooms S., & Martens, L. (2014). Context-aware recommendation through context and activity recognition in a mobile environment. Multimed Tools and Appl. Vol. 72 No. 3, pp 2925-2948.

[12] Chen, A. (2005). Context-Aware collaborative filtering system: Predicting the user's preference in the ubiquitous computing environment. In Location-And Context-Awareness, Springer Berlin Heidelberg, pp. 244-253.

[13] Koren, Y. (2008). Factorization meets the neighborhood: A multifaceted collaborative filtering model. In ACM SIGKDD, pp 426–434.

[14] Alhamid, M.F., Rawashdeh, M. H., Osman, A.L., Hossain, M.S. & El Saddik, A. (2013). Towards context-sensitive collaborative media recommender system. Multimed Tools and Appl., 1-30.

[15] Otebolaku, A.M. & Andrade, M. T. (2013). Recognizing high-Level contexts from smartphone built-In sensors for mobile media content recommendation. 14th IEEE International Conference on Mobile Data Management (MDM), pp.142-147.

[16] Bjelica, M. (2011). Unobtrusive relevance feedback for personalized TV program guides (2011). IEEE Transactions on Consumer Electronics, vol.57, no.2, pp.658-663.

[17] Lee, J. & Lee, J. (2008). Context awareness by case-based reasoning in a music recommendation system. Ubiquitous Computing Systems. pp 45-58.

[18] Otebolaku, A.M. & Andrade, M. T. (2014). A context-aware framework for media recommendations on smartphones. In proceedings of the European Conference on the Use of Modern Information and Communication Technologies(ECUMICT 2014), Gent, March 2014. 27th-28th of March 2014, Gent, Belgium, Published by Springer Lecture Notes in Electrical Engineering, edited by De Strycker, Lieven,Vol. 302, pp 87-108.

[19] Sara, B. M., Karypis, G., Konstan, J. & Riedl, J. (2001). Item-Based collaborative filtering recommendation algorithms. 10th International World Wide Web Conference, ACM Press, pp. 285-295.

[20] Zhu, H., Chen, E., Xiong, H., Yu, K., Cao, H., & Tian, J. (2014). Mining Mobile User Preferences for Personalized Context-Aware Recommendation. ACM Trans. Intell. Syst. Technol. 5(4).

[21] Oku, K., Nakajima, S., Miyazaki, J. & Uemura, S.(2006). Context-aware SVM for the context dependent information recommendation. In Proceedings of the 7th International Conference on Mobile Data Management, page 109.

[22] Oku, K., Nakajima, S., Miyazaki, J. & Uemura, S., Kato, H. & Hattori, F. (2010). A Recommendation System Considering Users' Past/Current/Future Contexts. ids.csom.umn.edu, pp. 3-7, [Online].

[23] Wang, X., Rosenblum, D. & Y. Wang, Y. (2012). Context-aware mobile music recommendation for daily activities. Proceedings of the 20th ACM international conference on Multimedia, October 29-November 02, 2012, Nara, Japan.