Interacting with Computers Advance Access published February 6, 2013

doi:10.1093/iwc/iws003

Predicting and Detecting the Relevant Contextual Information in a Movie-Recommender System

ANTE ODIĆ*, MARKO TKALČIČ, JURIJ F. TASIČ AND ANDREJ KOŠIR

University of Ljubljana, Faculty of Electrical Engineering, Tržaška cesta 25, Ljubljana, Slovenia
**Corresponding author: ante.odic@ldos.fe.uni-lj.si*

Context-aware recommender system (CARS) is a highly researched and implemented way of providing a personalized service that helps users to find their desired content. One of the remaining issues is how to decide which contextual information to acquire and how to incorporate it into CARS. While the relevant contextual information will improve the recommendations, the irrelevant contextual information could have a negative impact on the recommendation accuracy. By testing the independence between the contextual variable on the users' ratings for items, we can detect its relevance and impact on the feedback for the item consumed in that specific context. In this article, we propose several new theoretical concepts that should help deciding which information to use, as well as a methodology for detecting which contextual information contributes to explaining the variance in the ratings, based on statistical testing. The experiment was conducted on the real movie dataset that contains 12 different pieces of contextual information. We used two statistical tests with power analysis for the detection, and three contextualized matrix-factorization algorithms with slightly different reasoning for the prediction of ratings. The results showed a significant difference in the prediction of ratings in the context that was detected as relevant by our method, and the one that was detected as irrelevant, pointing to the importance of the power analysis and the benefits of the proposed method in the case of a small dataset.

RESEARCH HIGHLIGHTS

- We create a real context-aware movie-RS database.
- We provide theoretical concepts for contextual-information selection.
- We provide statistic-based method for relevant context detection.
- Relevant contextual information enhances rating prediction.
- Irrelevant contextual information degrades the rating prediction.

Keywords: personalization; user modeling; recommender systems; contextual information

Editorial Board Member: Paul van Schaik

Received 24 January 2012; Revised 21 September 2012; Accepted 26 September 2012

1. INTRODUCTION

With the huge amount of media content online, users are facing an everyday struggle to find their desired content in a reasonable time. Personalized services, defined as tailored services according to each user characteristic and preferences (Joung *et al*., 2009), are used when adapting a service to each specific user. One particular type of personalized service, recommender systems (RSs), is becoming an increasingly common part of online media-providing systems. Their main goal is to recommend relevant content by predicting ratings for items that have not been seen by a specific user (Adomavicius and Tuzhilin, 2005). Improving RS with contextual information, defined as information that can be used to describe the situation and the environment of the entities involved in such a system (Dey and Abowd, 1999), has been a popular research topic over the past decade. It was shown that an insight into situation

parameters can be used to improve the recommendation process, as well as in a number of different services (Toutain *et al.*, 2011). However, there are still a number of issues concerning the definition, acquisition, detection and modeling of this dynamic information (Yujie and Licai, 2010). It is not easy to decide which contextual information to use. For some systems the weather could be important (e.g. recommending a tourist destination), while for some it might be irrelevant (Dey, 2001). Contextual information that does not have a significant contribution to explaining the variance in the ratings could degrade the prediction, since it could play the role of noise (Baltrunas *et al.*, 2010). For that reason, we should be able to predict and later detect whether a specific piece of information should be acquired and used, or not.

In this article, we propose several theoretical concepts that define what contextual information is and how it differs from other types of information involved in the system. We also propose a methodology for the detection of contextual information that contributes to explaining the variance in the ratings. We validate theoretical concepts and our methodology on real data by using three different approaches to context-aware matrix factorization.

1.1. State of the art

The definition of contextual information that is commonly cited in related articles is the one proposed by Dey and Abowd (1999); Dey (2001): 'Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves'. By that definition, contextual information is much more dynamic and should not be confused with the general user information and the item metadata (Woerndl and Schlichter, 2007).

There is a variety of methods developed for context-aware RS (CARS). Adomavicius *et al.* (2005) presented the multidimensional recommendation model that adds context as the additional dimension in the classical $users \times items$ paradigm. The multidimensional approach was used by Hosseini-Pozveh (2009). The article also explains how to create a two-dimensional space by determining usage patterns under different contexts. A Bayesian network-based RS was proposed by Yap *et al.* (2007). The authors also tackled the problem of missing and erroneous context by harnessing the causal dependencies among the context variables. Three common ways of using the contextual information in an RS, i.e. context pre-filtering, post-filtering and context modeling, were explained by Adomavicius and Tuzhilin (2011). Baltrunas *et al.* (2009) used a pre-filtering method for the time context to improve music recommendations. The authors also tackled the problem of determining meaningful time partitions for pre-filtering, since the users can have different interpretations of time (morning, afternoon etc.). They compared different methods for determining these partitions. Su *et al.* (2010) used the contextual information for pre-filtering to produce a submatrix of the rating matrix, with only the most relevant users and items with similar context. Context similarity was calculated by Liu *et al.* (2010) to successfully tackle the problem of the increased data sparsity due to the context-reduction method or context filtering. Díez *et al.* (2010) used the random walks method to utilize social context in an RS. Rahmani *et al.* (2010) compared three approaches to a CARS: k-NN, linear-regression-based k-NN and inductive-learning programming (ILP).

Koren (2008) proposed a new approach to context-aware recommendations. The author combined both the neighborhood and the latent factor model. Time as a piece of contextual information was then introduced in the form of temporal dependencies of the ratings in the matrix-factorization technique (Koren, 2010). This approach led to first place in the Netflix prize competition (Koren *et al.*, 2009). Gantner *et al.* (2010) proposed, using the pairwise interaction tensor factorization (PITF), that Rendle (2010), used in tag recommendations, to predict which movies are rated in a specific time period. Baltrunas *et al.* (2010) extended the matrix factorization with an additional parameter that models the interaction of the context and the items.

A lot of different contextual information is being exploited in different domains. For example, emotional context was exploited by Gonzalez *et al.* (2007). Woerndl and Groh (2007) described ways that physical and social context can be used in an RS. Physical context was used to recommend mobile-device applications according to the location where they might be needed or that have been used by other users in a similar context. Social context was used to enhance the neighborhood creation in a collaborative RS. Social context was also used by Díez *et al.* (2010). Weather, mood and temperature, among others, were used in the tourist domain by Baltrunas *et al.* (2011).

From all the possible contextual information that can be acquired, it is necessary to decide which is important for a specific service. Adomavicius *et al.* (2005) used the paired t-test to detect which contextual information is useful in their dataset; however, in the dataset that we use for testing, all the contextual variables are categorical, thus the t-test could not be applied. Liu *et al.* (2010) used the χ^2 test for the detection of the relevant context; however, this test could be inappropriate for small datasets, i.e. for new systems and the cold-start problem, as we will explain later in the article. Baltrunas *et al.* (2011) conducted a context-relevance assessment to determine the influence of some contextual conditions on the users' ratings in the tourist domain, by asking users to imagine a given contextual condition and evaluate the influence of that condition. However, as they state, such an approach is problematic, since the users rate differently in real and supposed contexts (Ono *et al.*, 2009).

2. PROBLEM STATEMENT

While examining the state of the art, we identified several remaining issues concerning the definition of contextual information and the selection of the information to be used for that purpose. It is not always easy to predict which contextual information is important for a specific purpose. There are many situation parameters that can influence users' decisions in a more (location, social, working day/weekend) or less (weather, temperature) intuitive way. Not all contextual information should be used since the information that does not have a significant contribution to explaining the variance in the ratings could degrade the prediction. In addition, there is no need to spend resources to acquire and process data that are not relevant for the system.

Therefore, in the phase of designing a context-aware service, we must be able to predict which contextual information could contribute to the recommendation process, as a user-decision prediction. We will base this prediction on some of the important concepts regarding context that are included in the definition proposed in Dey and Abowd (1999), Dey (2001) and some additional concepts that we will introduce in the article. Once we have a reasonable amount of data collected, we can use statistical tests together with the power analysis to detect whether a specific piece of contextual information does in fact explain a part of the variance in the ratings. In this article we will use the Pearson's χ^2 test and the Freeman–Halton test for the detection. Once the relevant contextual variables have been detected, we can use them to enhance the recommendation outcome and provide the context-aware personalized service. Context variables can be used by contextualizing different parameters in the matrix-factorization algorithm, namely users' biases, item's biases and the users' feature vectors.

Our results show that the contextual variables detected as relevant significantly improve the rating prediction while the irrelevant ones lead to worse results than with the uncontextualized model. The results also point to the importance of the power analysis to avoid rejecting the variables that could improve the prediction.

The theoretical concepts regarding the contextual information that we introduce, along with the detection methodology that we propose, can help the acquisition and the inspection of the pieces of contextual information that will lead to the improvement in the rating prediction or other contextualized services.

3. THE PROPOSED DESCRIPTION OF THE CONTEXTUAL INFORMATION

In this section we propose the description of some theoretical concepts describing the contextual information. Only by understanding completely what the contextual information is, where and when to find it and how it behaves in time, during the user–item interaction, can we decide which information would be appropriate and how to approach the creation of the whole context-aware system architecture.

3.1. Item and interaction context

Contextual information does not only have to explain the situation of the user (Dey and Abowd, 1999). Apart from the user we can identify the *items' context* and the *user–item interaction's context*.

An item is an entity that is offered to the user for consumption, such as book, song, album, movie, restaurant, tourist destination etc. While different items are described by the metadata, one item can also have several different states, that can be described by that item's context. What the items' context is depends to a large extent on the type of the personalized service, i.e. items involved in the service, since the items can be very different from RS to RS. To explain how a single item can be different from situation to situation, let us consider the following examples. A specific movie is well defined by the metadata and the metadata is constant no matter what the consumption situation is. However, the media that contains, i.e. represents, the item can vary. A specific movie can be watched in different resolutions, video and audio qualities, screen sizes etc. These descriptions of the media from which the user consumes the item is the item's contextual information; it varies from situation to situation and can influence the user's rating. A book, for example, as a work of literature, also has a constant metadata; however, there are several editions, different cover types (e.g. paperback, hardback), different sizes (e.g. pocket size), there could also be missing pages, low-quality conditions of library books, price etc. Again, these descriptions of a specific object that represent the item contain the item's contextual information. Another example could exist in a restaurant RS. For example, the number of tables and type of a kitchen are the restaurant's metadata; however, the number of free tables currently available and the current menu or a special offer would be a piece of contextual information.

User–item interaction could also be considered as an entity that can contain its own context. There are a number of situation parameters that are related to the interaction that can influence the user. For example, the ordinal number of the interaction (e.g. the first interaction between the specific user and the item, *n*th interaction), interferences during the interaction, the quality of the interaction etc. This type of contextual information should not be disregarded since it could have a major impact on the users' decisions.

It is important to understand that even though these pieces of contextual information describe the states of different entities involved in the personalized service, they all influence the user's ratings.

3.2. Stages of the user–item interaction

Since the contextual information changes from situation to situation, it is important to acquire it at the right moment. To

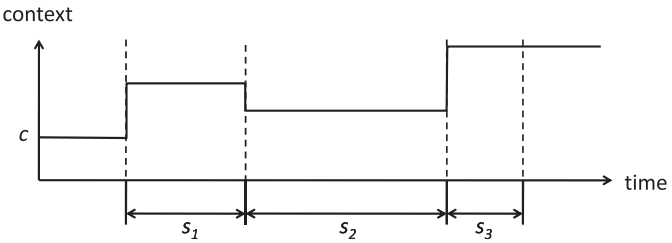

Figure 1. Interaction stages. Contextual information c is different during the decision stage s_1, consumption stage s_2 and the rating stage s_3.

ensure this, we define three distinct stages in time during the user–item interaction: *decision stage*, *consumption stage* and *rating stage*.

The first stage is the decision stage. During this stage, a user is trying to find the most appropriate content. The following consumption stage is the period in which the user consumes the content. During this, stage different factors are influencing the user's satisfaction with the content. Last is the rating stage where the user rates the content, based on the satisfaction and the impression from the consumption stage.

The important thing to have in mind is that these stages are not necessarily in the same time period, and since the contextual information is highly dynamic, wrong contextual information could be obtained (Fig. 1). Since the task of a context-aware RS is to predict which item would be the most appropriate for a specific user to consume in a specific situation, the main interest is in the contextual information from the consumption stage. For example, if a user selects a movie on Monday, watches it on Tuesday and rates it on Wednesday, we are interested in the contextual information obtained on Tuesday, when the user consumed the item. This is important because the timestamp for the ratings in the existing RS datasets is often from the rating stage and does not provide us with the information about when the user consumed the rated item, as we will explain in Section 5.1.

3.3. Time span of the user–item interaction

To make sure that the acquired contextual information is appropriate, we also have to take the duration of the user–item interaction into consideration. This is because some pieces of contextual information are more dynamic than others (e.g. an emotional state changes much faster than the days in the week or the seasons). For this reason we define three different user–item interaction periods: *long-term interaction*, *medium-term interaction* and *short-term interaction*. In addition to that, if the contextual information has a constant value during the whole consumption stage, we call it a *single-context value* stage. In contrast, if the context value changes during the consumption stage, we call it a *multiple-context value* stage.

In the case of the *long-term interaction* period, for most contextual information the consumption stage is a *multiple-context value* stage, and the values of most of the contextual information are different between the interaction stages. For example, let us consider context-aware recommendations for books. A book is consumed over a long period of time. The user changes different locations, weather conditions, emotional states and days while consuming this type of item. In addition, the context is certainly different between the user–item interaction stages.

In the case of the *medium-term interaction* period, for most contextual information the consumption stage is a *single-context value* stage; however, the values of most of the contextual information are still different between the interaction stages. For example, during the consumption of a movie, the user's surroundings do not change much, the movie is consumed on one day, one location, constant weather etc. Only some of the contextual information change several times, like the emotional state, for example.

In the case of the *short-term interaction* period, for all the contextual information the consumption stage is a *single-context value* stage and the values of the contextual information are the same between the interaction stages. For example, interacting with music, Youtube videos and images are considered a *short-term interaction* period.

Figure 2 shows the example of the three different possible interaction periods. On the context axis, two different contextual variables are shown, c_1, whose states are longer lasting (e.g. social state) and c_2, which changes its states quickly (e.g. emotional state). For the *long-term interaction*, for example, reading a book, we can see how context values change several times during the consumption (*multiple-context value* stage), and how they are different during different stages. Hence, if

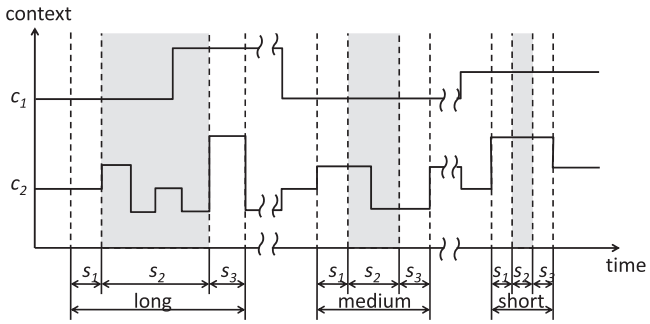

Figure 2. *Long-term*, *medium-term* and *short-term interaction*. During the *long-term interaction* period the context values are different between stages, and vary during the consumption stage s_2. During the *medium-term interaction* period the c_2 context value is different between stages, and varies during the consumption stage s_2. During the *short-term interaction* period the context values are the same between the stages and constant during the consumption stage s_2. These periods are independent of each other. The consumption stage s_2 is marked gray in each interaction period.

we would gather the contextual information during the rating stage, we would have the wrong idea about the context while the user was consuming the item. A similar situation occurs for the *medium-term interaction*, for example, watching a movie; however, c_1 is constant during all the stages. For a *short-term interaction*, for example, listening to a song, both contextual variables have constant values during all the stages.

Of course, this concept varies from application to application, but it should not be overseen because the acquired contextual information could be inadequate. Tkalčič *et al*. (2011) also divided the users' interaction with an RS into three stages based on the role that emotions play in the interaction.

3.4. Levels of variation

While trying to explain the variance in the ratings we may use different types of information on two levels: *the population level* and *the user level*. On the *population level*, we inspect the rating behavior of the population (i.e. how users rate the items). On the *user level*, we inspect the rating behavior of the user (i.e. how a user rates the items). On both levels we have a type of information that is constant and one that varies.

The variance on the *population level* could be explained through general user information. Users from different age groups, with different levels of education and demographic differences, may rate the same item differently. The constant population information is the same for the whole population using the system, for example, users are all humans that know how to use the internet. It is clear that such an obvious piece of information is the same for all the users and is unimportant. On the population level we thus have variable ratings for a fixed item from different users.

On the *user level*, the general user information becomes constant. For one user, his sex, education, residence etc., is constant and thus does not explain the variance of his ratings. Contextual information, however, is dynamic and varies from situation to situation. On the *user level*, a different social state, location, weather, emotional state, mood and other contextual information could explain a part of the variance in the ratings. One user could rate the same item differently on different occasions, which could be explained by the contextual information. We thus have variable ratings for a fixed item in different situations, i.e. a different context.

3.5. Predicting relevant context

With these concepts in mind, we can now predict which contextual information could be relevant for a specific system.

If a specific context were to be constant due to the specific system, then that context is irrelevant and should not be used. For example, if users will always consume items at home, then the location as a piece of contextual information is irrelevant for that specific service. Dey (2001) provided the example of not using weather as contextual information for indoor-only applications.

If in a long- or medium-term user–item interaction the interaction stages are not at the same moment in time we have to make sure that the contextual information is obtained in such a way that it describes the situation during the consumption stage. For example, weather obtained during the rating stage is irrelevant.

In a long- or medium-term user–item interaction, when a specific context changes several times during the consumption stage, it has to be modeled. For example, if a user is consuming a book for a long period of time (e.g. 1 month) the *weather* variable should describe what the weather was mostly like during the whole consumption stage.

When the context is different between the decision and consumption stages, we have to 'predict' what will the context value be during the consumption stage, in order to provide a contextualized service. For example, if a user is using a system to get a tourist-destination recommendation 1 month in advance, the system should not provide a recommendation based on the present situation (spring, cloudy, average temperature 22°C), but on the situation at the time of the trip (summer, sunny, average temperature 28°C). If it is impossible to 'predict' the context value from the consumption stage, such context is irrelevant for a real application.

To summarize, based on the explained theoretical concepts we add the following requirements to the existing definition of contextual information. By using these requirements we should be able to predict whether a certain piece of information is contextual and could be used in the specific service.

(i) Information is contextual and could be relevant in the specific service:
- if it varies at the user level, i.e. varies for a fixed user from situation to situation
- if it describes the situation during the consumption stage of the user–item interaction, i.e. provides the information about the consumption circumstances
- if in the case that it has multiple values during the consumption stage, it can be modeled in the appropriate way
- if it is possible to predict the consumption-stage context value at the moment of preparing a recommendation

(ii) Contextual information is in fact relevant if it has a significant contribution to explaining the variance in the ratings.

4. STATISTICAL MEASURES FOR RELEVANT-CONTEXT DETECTION

In this section, we present the basic statistical methodology used to identify the relevant context variables out of a set of

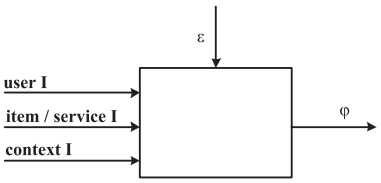

Figure 3. A symbolic representation of the user-decision process.

potentially contextual variables (those expected to be contextual from a basic understanding of the problem we analyze, and the requirements from Section 3.5). As already indicated in Section 2, the reason for contextual-variable identification is the fact that a given variable can improve, be neutral, or aggravate the prediction of a user rating.

4.1. Basic notations and reasoning

Assume that the decision $\varphi(u, h, c)$ (such as the rating a user has assigned to a given content after the consumption) of a user $u \in \mathcal{U}$ on an item $h \in \mathcal{H}$ in the context $c \in \mathcal{C}$ is an additive sum of contributions

$$\varphi(u, h, c) = \varphi_u(u, h, c) + \varphi_h(u, h, c) + \varphi_c(u, h, c) + \varepsilon, \quad (4.1)$$

where φ_u is the contribution of the user model UM(u), φ_h is the contribution of a content item or service metadata MD(h), φ_c is the contribution of a context to the user decision φ and ε is a random variable modeling part of the user decision (Fig. 3).

Each of the listed contributions is estimated from several variables containing the corresponding information. The variables are indexed by the index $1 \leq i \leq m$, those related to the user model are denoted by v_i^u, those related to the content metadata are denoted by v_i^h and those related to the context are denoted by v_i^c.

The basic reasoning behind the symbolic representation of the user decision process modeled by $\varphi(u, h, c)$ is that only a part of its variability can be explained by the contributions φ_u and φ_h. An unexplained variability of $\varphi(u, h, c)$ is modeled by a random variable. The reason for this is the nature of the human decision-making process, which is very complex and dependent on many other factors besides the content item and user's past behavior recorded in his user model. The assumption is that an additional part of the user-decision variability can be explained by the user context c captured by the variables v_i^c and modeled by φ_c. The rest of the user-decision $\varphi(u, h, c)$ variability is modeled by a random variable, see Equation (4.1).

Clearly, a variable that has no variability for a given user cannot provide any additional information about his decisions and cannot be relevant context. On the other hand, the high variability can be inconsistent with the rating. Therefore, if the variability of the variable is low, it is not a relevant contextual information, if the variability is high it might or might not be a relevant contextual information. Furthermore, a variable that does not explain a significant part of the user-decision variability φ cannot provide any relevant contextual information. Therefore, the two ways to identify relevant context variables are:

(i) Measure the variability of the variable v_i^c;
(ii) Measure the part of the variability of the user decision φ that can be explained by the variable v_i^c.

Since the approach (i) gives the necessary condition for a variable to be contextual and the approach (ii) gives the sufficient condition, in this paper we concentrate on the approach (ii). Note that the approach (i) does not use a user-decision variable as an input but only measures the variability of the variables, regardless of user decisions.

The application of both of the above-listed approaches depends on the analyzed variables type. When they are of the interval or proportional type, it is based on the Pearson correlation coefficient (factor analysis and regression analysis). But in the real world the candidate variables are typically of the categorical or ordinal type, which requires a modified approach. Details are given in Section 5.2.

5. MATERIALS AND METHODS

In this section we provide a description of the datasets and the methods used in our experiment. First, we used the requirements defined in Section 3.5 to predict which information is potentially contextual in the movie domain. Based on this, we prepared the data acquisition. Data acquisition is basically the natural use of the system in which the users rate the items, i.e. movies they have consumed, as explained in Section 5.1. Once the substantial amount of data was acquired, we employed the relevant-context-detection procedure (5.2) in order to determine which pieces of contextual information are in fact relevant. Based on the detection results, we selected those pieces of contextual information which were detected as relevant and used them in the rating prediction, i.e. RS. From the detection results, we could also modify the data acquisition stage to stop acquiring the irrelevant pieces of contextual information. In this article we used the RS to evaluate the detection method. We used two statistical tests and three different context models in the RS. The diagram of the whole procedure is shown in Fig. 4.

5.1. Dataset

Initially, we examined some of the available, existing RS datasets that are commonly used. First, we examined the Moviepilot and Filmtipset datasets that were used in the CAMRA challenge (2010 and 2011). In parts of these datasets that we could acquire, the only contextual information was the one that could be derived from the timestamps. Social connections between the users and other similar information in these datasets are general user information and do not vary at the user's level, thus they are not potential contextual information

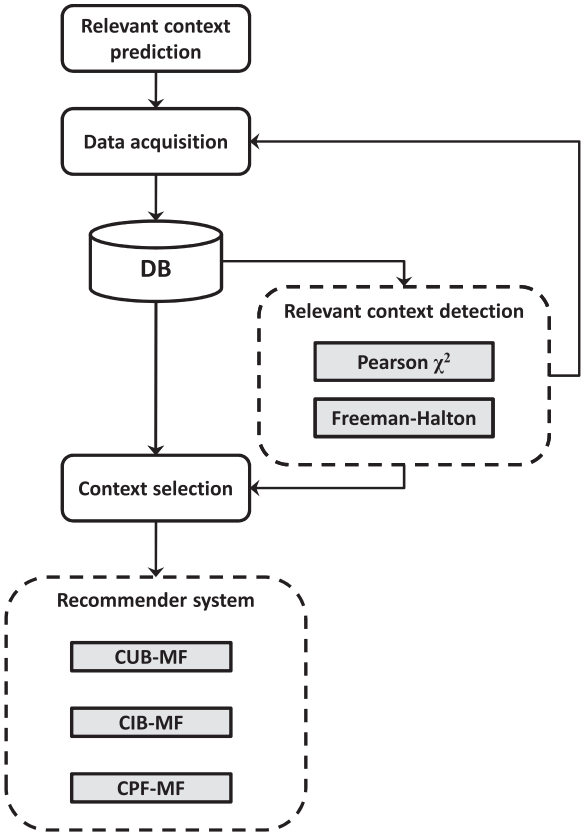

Figure 4. Architecture of the context-aware RS. Relevant context prediction is used to decide which contextual information to acquire. Relevant context detection is used to detect which contextual information is relevant in our RS, based on which we can (i) modify the data acquisition and (ii) select the relevant features(contextual information). We use the RS to evaluate the detection.

according to the definition from Dey and Abowd (1999), Dey (2001) and our requirements explained in Section 3.5. Also, after examining the timestamps in these datasets, we found that most users have all the timestamps from a relatively short period of time, usually most of them from the same day. This points to the fact that users provide most of the ratings at once. This means that these timestamps are from the rating stage and that we do not know when the user consumed the item. As we explained earlier, we are interested in how a specific item suited the specific user in a specific situation, i.e. what was the context during the consumption stage, not the rating stage.

We also examined the Yahoo music dataset used in the Yahoo KDD cup in 2011. We concluded that since music has a *short-term interaction* period, these timestamps could also be related to the consumption stage (user listens to a song and then rates it immediately after). However, this was also the only potential piece of contextual information.

Since we were interested in inspecting several contextual variables, we decided to create a dataset containing several potential pieces of contextual information from the consumption stage. Therefore, we created an online application for rating movies (*www.ldos.si/recommender.html*). Our users are employing the application to track the movies they watched, obtain the recommendations and browse movies.

In addition to a rating for a movie consumed, users fill in a simple questionnaire created to explicitly acquire the contextual information describing the situation during the consumption stage of the user–item interaction. The questionnaire was designed in such a way that it is simple and not time consuming for a user to provide the contextual information. Users are instructed to provide the rating and contextual information immediately after the consumption, so that we can be sure that the data are registered during the consumption stage, and that the ratings are not influenced by any other factor (e.g. discussing the movie with others, observing the average movie score on the internet etc.) between the consumption and rating.

The users' goal for rating movies is to improve their profile, express themself and help others, according to Herlocker *et al.* (2004).

Our context-movie dataset (LDOS-CoMoDa) that contains contextual information has been in development since 15 September 2010. It contains three main groups of information: general user information, item metadata and contextual information.

The general user information is provided by the user upon registering in the system. It consists of: user's age, sex, country and city. This information is constant at the user level (Section 3.4) and thus cannot be used as a piece of contextual information. Item metadata is being inserted into the dataset for each movie rated by at least one user. The metadata describing each item is: the director's name and surname, country, language, year, three genres, three actors and budget.

The decision about which contextual information to collect was made according to the theoretical concepts described in Section 3. We did not want to infer the *time* (e.g. morning, afternoon, evening, night) as a context variable from the timestamps since different users might have different habits and thus different definitions of what *morning*, *afternoon*, *evening* and *night* are (Baltrunas *et al.*, 2009). Therefore, we let users chose the right 'value' of time from the list of possible time classes, as they did for all the other contextual information. We were also interested in the emotional state of the user as contextual information, since Tkalčič *et al.* (2010b) showed the positive impact of affective parameters in an RS. However, for the emotional state as the contextual information in the movie RS, the consumption stage is a *multiple-context value* stage, as explained in Section (3.3). For that reason we decided to collect the emotional state that was dominant during the consumption (*domEMO*) and the emotional state at the end of the movie (*endEMO*). We also collected the *mood* as the piece of contextual information. Ekman and Davidson (1994) state that the moods last much

Table 1. Contextual variables in our dataset.

Contextual variable	Description
time	morning, afternoon, evening, night
daytype	working day, weekend, holiday
season	spring, summer, autumn, winter
location	home, public place, friend's house
weather	sunny/clear, rainy, stormy, snowy, cloudy
social	alone, partner, friends, colleagues, parents, public, family
endEmo	sad, happy, scared, surprised, angry, disgusted, neutral
dominantEmo	sad, happy, scared, surprised, angry, disgusted, neutral
mood	positive, neutral, negative
physical	healthy, ill
decision	user's choice, given by other
interaction	first, *n*th

Table 2. Basic dataset statistic.

Number of users	89
Number of items	946
Number of ratings	1611
Average age	27
Number of countries	6
Number of cities	16
Max ratings of a single user	220
Min ratings of a single user	1

longer than the emotions. Furthermore, in our case, mood is not necessarily triggered by the item that is being consumed, but describe the overall mood of the user at the time of the consumption.

Among other contextual information, we also wanted to inspect the impact of the context variables that describe the user–item interaction, as we explained in Section 3.1. For that purpose we chose two different pieces of contextual information: *decision* (did the user decide on which movie to watch on his own or was the movie proposed to him by a TV program, friend, RS etc.) and *the interaction number* (did the user watch the movie for the first time).

All the contextual variables and values describing the consumption stage of the user–item interaction that we collected are listed in Table 1. Table 2 contains some basic statistics about the data.

Another important database property is the amount of ratings provided by each user and the amount of ratings provided for each item. If most of the users have provided only a small amount of ratings or if there is only a small amount of ratings per each item (also known as the *long tail* in the ratings distributions for items and users (Park and Tuzhilin, 2008)), it is hard to train a

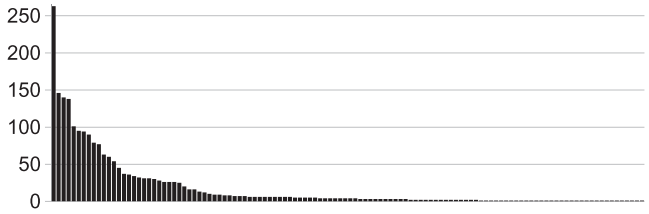

Figure 5. Number of ratings per user. Heights of the bars represent the number of ratings provided by each user in the database.

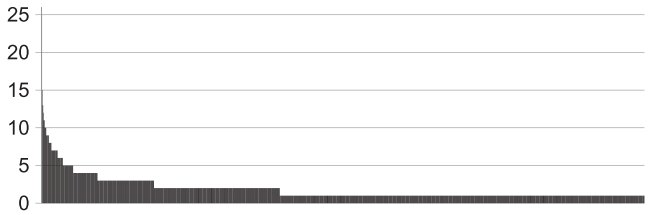

Figure 6. Number of ratings per item. Heights of the bars represent the number of ratings provided for each item in the database.

model. This is especially true for some types of contextualized methods, like pre-filtering Adomavicius and Tuzhilin (2011), since the number of ratings provided in a specific context is even lower than the overall number of ratings.

Figures 5 and 6 show the amount of ratings provided by each user and the amount of ratings provided for each item, respectively.

5.2. Detecting relevant context

For the detection of the relevant contextual variable, we used statistical hypothesis testing with a power analysis, as described in this section. For the reasoning behind the approach (i) and approach (ii), see Section 4.1.

5.2.1. Context-variable identification using a variable variance (approach (i))

A variability of a random variable v is typically computed by a covariance formula $\eta(v) = \sqrt{\mathrm{Cov}(v, v)}$. Unfortunately, contextual variables are typically not numeric but categorical or at most ordinal. Besides, straightforward adaptations of this formula using association coefficients among categorical variables such as contingency coefficients are also not applicable since it does not have all required properties. Therefore, we have to use other variability measures. Some of them are presented in the subsequent paragraphs.

Kader and Perry (2007) proposed the unalikeability as a measure of how often observations differ from one another. If $x_1, x_2, \ldots, x_n$ are the observations of a variable v^c, variability as unalikeability is calculated as:

$$\eta_u(v^c) = \frac{\sum_{i \neq j} c(x_i, x_j)}{n^2 - n} \tag{5.1}$$

where v^c is a contextual variable, x_i and x_j are the observations of the variable v^c, n is the number of all observations of the variable v^c, and $c(x_i, x_j) = 0$ if $x_i = x_j$, and $c(x_i, x_j) = 1$ if $x_i \neq x_j$.

Another approach is to calculate the variability as the entropy of the observations, by the formula:

$$\eta_H(v^c) = -\frac{1}{\log_2 M} \sum_{i=1}^{M} \frac{n_i}{n} \log_2 \frac{n_i}{n} \tag{5.2}$$

where M is the number of possible values of a contextual variable v^c, n_i is the number of occurrences of variable v^c taking value i, and n in the number of all observations of a variable v^c. $1/\log_2 M$ is a normalizing constant which ensures that the variability is between zero and one.

Variability as a sample variance is calculated as:

$$\eta_v(v^c)^2 = \sum_{i=1}^{M} n_i |v_i - \bar{v}|^2 \tag{5.3}$$

where v_i is the value of the categorical class i and $\bar{v}$ is the mean of all the observations.

A statistical test for significant variations could be applied here but for simplification we simply order variables according to their variability and identify a selected number of most variable variables as candidates for contextual variables.

Note that these variability measures possess all properties we require in order to apply the approach (i), devised to eliminate variables with low variability from the contextual variables candidates list.

Using these three methods, we calculated the variability of each potentially contextual variable on the whole population, both for all the ratings and for each fixed rating separately. We also calculated the variability for each potentially contextual variable on the user level, i.e. for each fixed user. The reasoning behind this is the following. If the variability of a variable v_i^c is low on the population level, it can surely be rejected, as irrelevant contextual information. However, if the variability is high, it is still possible that for a given user it could be low. For example, a specific user always watches the movies at night. For him the variable *time* is not relevant contextual information.

5.2.2. *Context variable identification using the explained variance of a user decision* (*approach* (ii))

The independence was tested between each contextual variable and the ratings. The null hypothesis of the test states that the two variables are independent. The alternative hypothesis states that they are dependent. If we successfully reject the null hypothesis, we conclude that the contextual variable and the rating are dependent and thus the contextual information is relevant.

Since the potentially relevant contextual variables under investigation are typically categorical or at most ordinal, the association among the variables (contextual and rating) is measured using association coefficients and pertained significance tests for the categorical variables. The A-priori and the post-hoc power analyses are of key importance here.

The only association coefficient applicable for all involved variables (some are categorical) is the Cramer's coefficient V and the pertained significance test for association is the χ^2 test. If the candidate context variables are of the ordinal type, the same reasoning is applicable when the Cramer's coefficient V is replaced by the Kendall rank correlation coefficient τ and the significance test for zero hypotheses $H_0 = [\tau = 0]$ is applied. However, in the case of an ordinal variable with a small number of values, it can be treated as a categorical variable. Liu *et al.* (2010) also used the χ^2 test for relevant-context detection; however, this test is not appropriate in our dataset due to Cochran's rule, which states that at least 80% of the expected cell count in a contingency table should be five or more and that no expected cell count should be less than 1. In small datasets (e.g. during the cold-start phase) and especially when one of the variables tested has a larger number of possible values, Cochran's rule is not satisfied and the detection cannot be achieved. For that reason, we propose using the Freeman–Halton test, which is Fisher's exact test extended to $n \times m$ contingency tables, since the Fisher test does not depend on the sample size (Agresti, 1992).

From the four possible outcomes of the hypothesis testing, two outcomes, i.e. the type-I and type-II errors, would mean that the relevance of the contextual information was not detected correctly. A type-I error would mean that the contextual variable was irrelevant, but was detected as relevant. In other words, it would be used as relevant information when in fact it does not explain a part of the unexplained variance in the rating. A type-II error would mean that the contextual information was in fact relevant, but was detected as irrelevant. In other words, the information that does explain a part of the unexplained variance in the rating was overseen. For that reason we are especially interested in observing the probability of a type-II error in our hypothesis testing. The probability of a type-II error occurring is called the false-negative rate β. Therefore, the probability of obtaining a result from the statistical test that will allow the rejection of the null hypothesis, if the null is false, is called the statistical power and is equal to $1 - \beta$.

Once the p value (p) and the statistical power ($1 - \beta$) are calculated for a contextual variable, the probability that the contextual variable is relevant is determined by the following equation:

$$P[A] = \begin{cases} 1 - \beta, & p < \alpha \\ \beta, & p \geq \alpha \end{cases}$$

where A stands for a *contextual variable that is relevant*, which means that it has a significant contribution in explaining the variance in the ratings, and α is the significance level of the test.

In the case when $p \geq \alpha$, if the statistical power is high, we conclude that the contextual variable is not relevant; however, if the statistical power is small, we do not conclude that the

context is not relevant. This principle is independent of the test that we use.

We also conducted an a-priori power analysis to compute the required sample size, given the significance level, desired power and effect size.

The a-priori power analysis and post-hoc power analysis were conducted according to Cohen (1988) and Faul (2009). The post-hoc power analysis for the Freeman–Halton test was made using the Monte-Carlo method, and was implemented using the *R* software environment (http://www.r-project.org/).

5.3. Context-aware recommendation

In order to evaluate the impact of the contextual information detected as relevant and the one detected as irrelevant on the ratings prediction, we used three different contextual models for predicting the ratings. They are all based on the matrix factorization as a collaborative-filtering algorithm (Koren, 2008); however, the way the contextual information is used in them differs. The purpose of these different approaches is to inspect how the detection of the relevant context influences each parameter that can be contextualized. We therefore contextualize the user's bias, the item's bias and the user's preferences, represented with the latent feature vector.

To inspect the impact of a certain piece of contextual on the rating prediction, we trained each model for each piece of contextual information separately, i.e. using only a single piece of contextual information at the time. In that way we acquire the root mean square error (RMSE) score from each model utilizing each piece of contextual information.

We used the following equation and notations for the matrix factorization:

$$\hat{r}(u,i) = \mu + b_i + b_u + \mathbf{q}_i^{\mathrm{T}} \cdot \mathbf{p}_u \tag{5.4}$$

where $\hat{r}(u,i)$ is the predicted rating from a user u for the item i, μ is a global ratings' bias, b_u is a user's bias, b_i is an item's bias, $\mathbf{q}_i$ is an item's latent feature vector and $\mathbf{p}_u$ is a user's latent feature vector. $\hat{r}$, μ, b_u and b_i are scalars, and $\mathbf{q}_i$ and $\mathbf{p}_u$ are vectors. The contextual variable in the following equations will be denoted by c.

The system learning procedure is defined as an optimization problem:

$$\min_{p_*,q_*,b_*} \sum_{r\in K} \Big[(r(u,i) - \mu - b_i - b_u - \mathbf{q}_i^{\mathrm{T}} \cdot \mathbf{p}_u)^2 + \lambda \sum_{r\in K} (b_i^2 + b_u^2 + \|\mathbf{p}_u\|^2 + \|\mathbf{q}_i\|^2) \Big]$$

where λ is the constant that controls the regularization and $K = \{(u,i) \mid r(u,i) \text{ is known}\}$. The learning rate of the training was set to 0.00001 and $\lambda = 0.005$. These values were determined in the factorial design, by observing the changes in the response produced by the changes in the value of these two factors.

We calculated the users' and items' feature vectors using the gradual descent method (Koren, 2008). The items' latent features calculated in this way measure more or less obvious dimensions describing the items (e.g. comedy versus drama, amount of action), or even completely uninterpretable dimensions (Koren *et al.*, 2009). The users' latent features measure how much the user likes movies that score high on the corresponding movie factor (Koren *et al.*, 2009). The items' biases were calculated as the difference between the mean rating for a specific item and the global bias μ. Similarly, the users' biases were calculated as the difference between the mean rating for a specific user and the global bias μ. Global bias is calculated as an average of the ratings in the training set.

We used the RMSE as the evaluation measure for the predicted ratings.

Contextualized users' biases with the matrix factorization (CUB-MF) model uses the contextual information for the contextualized users' biases. Only the users' biases are context dependent. This approach follows the idea that the users' rating behavior is different on different occasions, though their tastes are the same. The matrix factorization in CUB-MF was made using the following equation:

$$\hat{r}(u,i) = \mu + b_i + b_u(c) + \mathbf{q}_i^{\mathrm{T}} \cdot \mathbf{p}_u \tag{5.5}$$

The contextualized users' biases were calculated for each possible value of the piece of contextual information used in the model. For example, if we are using *day type* as the piece of contextual information, we have three categorical values, *working day*, *weekend* and *holiday*, thus we calculate three biases: *working-day* bias, *weekend* bias and *holiday* bias. During the model training, these biases are calculated by using only those ratings with the associated context value. For example, *weekend* biases were calculated only from the ratings provided during the weekend context. Similarly, when predicting the rating that occurred during context value 'weekend', we use only the calculated *weekend* bias.

Contextualized items' biases with the matrix factorization (CIB-MF) approach uses the contextual information for contextualized items' biases, so only the items' biases are context dependent. The reasoning behind this approach is that the items are generally rated differently on different occasions. The matrix factorization in CIB-MF was made using the following equation:

$$\hat{r}(u,i) = \mu + b_i(c) + b_u + \mathbf{q}_i^{\mathrm{T}} \cdot \mathbf{p}_u \tag{5.6}$$

The contextualized items' biases were calculated in the same manner as the users' biases in the CUB-MF model.

For the last type of CARS we chose combining the pre-filtering method with the matrix factorization (CPF-MF). This approach assumes that the users' preferences differ from situation to situation. Since the users' feature vectors are context

dependent in this approach, the focus is more on how a specific user likes movies that score high on the corresponding movie factor in a specific context. The item's feature vector $\mathbf{q}_i$ is not context dependent, since the items do not differ on different occasions. The matrix factorization in the CPF-MF method was made using the following equation:

$$\hat{r}(u,i) = \mu + b_i + b_u + \mathbf{q}_i^{\mathrm{T}} \cdot \mathbf{p}_u(c) \tag{5.7}$$

The contextualized users' feature vectors were calculated in the same manner as the users' biases in the CUB-MF model.

We also calculated an RMSE score for the two baseline predictors, the basic matrix factorization without context (MF) (Equation 5.4), and the simplest method that calculates the predicted rating as an average item's rating (AVG).

If the contextualized parameter could not be trained for a specific context value due to data sparsity (for example, user did not consume any movie during snowy weather), the uncontextualized parameter was used during the prediction of that specific rating. In the extreme case, if the contextualized parameter could not be calculated for any context value, the prediction result would be the same as with the uncontextualized model (MF).

On the acquired data, we applied the 10-fold validation strategy. Each fold consists of the training set and the test set, containing 90 and 10% of the dataset entries, respectively (each dataset entry consists of the user id, item id, rating and the context values). Since there are users and items with only a few ratings (Figs 5 and 6), it can occur that for a specific user in the test set there are no ratings in the train set. However, we did not want to delete such users or items from the dataset, since these can also have an impact on the training (as in the real online setting). Therefore, while creating the test and train sets for each fold, we dealt with the sparsity in the following way: for each user or item in the test fold, there had to be at least five ratings in the train fold. In the real online setting, this would mean that the rating predictions (i.e. recommendations) can be achieved for those users who have rated at least five items, and for the items that have been rated at least five times.

On each fold we made the rating prediction by a model contextualized with each piece of contextual information separately. In this way we can compare how each piece of contextual information performed, and whether the relevant context led to better prediction than the irrelevant one, on each fold. On each fold we also use the baseline predictors MF and AVG.

We assume that the model that employs pieces of contextual information detected as relevant should lead to better results than the model that employs pieces of contextual information detected as irrelevant. Furthermore, the model that employs pieces of contextual information detected as relevant should lead to better results than the uncontextualized model. Finally, since the irrelevant context can play the role of noise, a model that employs pieces of contextual information detected as irrelevant should lead to worse results than the uncontextualized model. However, in the case of sparse contextual information, the results could be similar to those of the uncontextualized model. To summarize:

(i) relevant context RMSE should be lower or equal than the irrelevant context RMSE
(ii) relevant context RMSE should be lower or equal than the baseline RMSE
(iii) irrelevant context RMSE should be greater or equal than the baseline RMSE.

6. RESULTS

6.1. Detection

Using the methods described in Section 5.2, we tested each contextual variable in our dataset to detect which one is relevant. The significance level of our test was $\alpha = 0.05$.

In the a-priori power analysis, we used the effect size $w = 0.2$, $\alpha = 0.05$ and the power $(1 - \beta) = 0.95$. For the variables with the lowest degrees of freedom $df = 4$, the required sample size calculated was 456 samples. For the variables with the highest degrees of freedom ($df = 24$), the required sample size calculated was 820 samples.

A post-hoc power analysis gives us the achieved power, with an insight into the real effect size.

Variable variability (approach (i)) did not detect any contextual variable as irrelevant. Variabilities of all the contextual variables in our dataset were high and thus all the variables were detected as potentially relevant by approach (i). Therefore, approach (i) did not prove to be useful in this dataset.

Table 3 contains the results of the contextual information detection (approach (ii)), for each variable and both tests, according to the rules in Section 5.2.2. In the 'context' column, the names of the six context variables are written in bold characters. Those are the variables that were detected as contextual. For (*endEmo*, *domEmo*, *mood*, *physical*, *decision* and *interaction*), the p value was less than the significance level. *Location* and *dayType* are variables that could not be declared as irrelevant, since the calculated power was low (written in italic characters). The other four variables, time, season, weather and social, are rejected (i.e. irrelevant contextual information). We thus classified each of the 12 pieces of contextual information into one of the three groups: *relevant*, *irrelevant* and *unable to classify*.

For the Pearson χ^2 test, the Cochran's rule was not satisfied for these three variables: *weather*, *social* and *end emotion*.

6.2. Evaluation of the proposed context-variable selection mechanism

By using the models described in Section 5.3, we made the rating prediction while employing each piece of contextual information separately.

Table 3. Context-detection results.

	Freeman–Halton		χ^2		
Context	*p*	Power	*p*	Power	Cochran
time	0.33	0.64	0.33	0.67	Yes
season	0.15	0.76	0.14	0.79	Yes
weather	0.11	0.84	–	–	No
social	0.08	0.92	–	–	No
dayType	0.87	0.19	0.86	0.24	Yes
location	0.86	0.19	0.90	0.21	Yes
endEmo	<0.01	>0.99	–	–	No
domEmo	<0.01	>0.99	<0.01	>0.99	Yes
mood	<0.01	0.99	<0.01	>0.99	Yes
physical	0.03	0.74	0.05	0.69	Yes
decision	<0.01	0.95	<0.01	0.94	Yes
interaction	<0.01	>0.99	<0.01	>0.99	Yes

Due to the long tail in the rating per item distribution, i.e. small number of rating per item in LDOS-CoMoDa dataset, the results from the CIB-MF model were inconclusive. Since the contextualized items' biases could not be calculated for many context values, uncontextualized biases were used, making all the results similar to the uncontextualized model, as explained in Section 5.3.

Since the users' feature vectors, contextualized in the CPF-MF model, explain only a small part of the variance in the ratings, the differences between the RMSE values from different pieces of contextual information were too small to conclusively show the impact of the relevant and irrelevant pieces of contextual information on the rating prediction.

The results from the CUB-MF model are shown on the boxplot in Fig. 7.

The overlap in the boxes in Fig. 7 should not be interpreted as the conclusive evidence that there is no significant difference between the results since the results are paired, i.e. results are achieved from different methods on the same 10 folds. Since the results from the 10 folds are not normally distributed, we used the Wilcoxon signed-rank test to find whether the differences between the results for the relevant and irrelevant contexts are significant. The results are presented in Table 4.

In order to visualize the results form Table 4, the differences in the RMSE between the CUB-MF model contextualized

Table 4. Significance of the performance difference between contextual variables. Table contains the p-values of the Wilcoxon signed-rank test between the irrelevant and both relevant and unclassified contextual variables.

	Time	Season	Weather	Social
dayType	0.0645	0.0273	0.6953	0.2754
location	0.1055	0.6953	0.0020	0.0645
endEmo	0.0020	0.0020	0.0020	0.0020
domEmo	0.0020	0.0020	0.0020	0.0020
mood	0.6953	0.3750	0.0195	0.8457
physical	0.0020	0.0020	0.0020	0.0059
decision	0.0020	0.0098	0.4316	0.0137
interaction	0.0059	0.0020	0.5566	0.0195

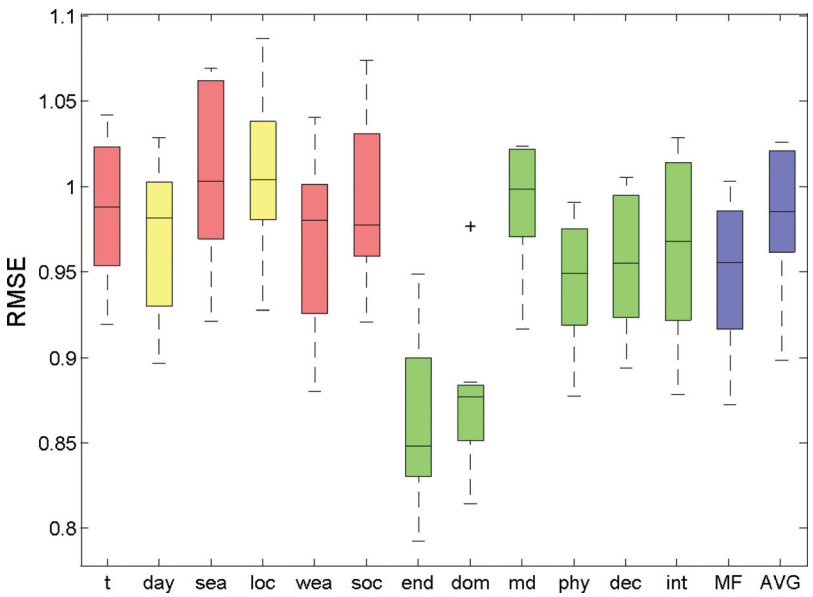

Figure 7. Boxplot of the 10-fold validation with the CUB-MF model. Boxplots represent the distribution of the RMSE results on the 10 folds from the CUB-MF model contextualized by each of the irrelevant context (red), relevant context (green), context that could not be classified (yellow) and the baseline predictors (blue).

by each irrelevant context variable and the CUB-MF model contextualized by both each relevant and unclassified context variables were calculated on each fold as $RMSE_{irrelevant} - RMSE_{relevant}$ and $RMSE_{irrelevant} - RMSE_{unclassified}$. If the difference is negative, the irrelevant variable led to better results than the relevant or the unclassified variable. If the difference was positive, the relevant or the unclassified variable led to better results than the irrelevant variable. The distributions of these differences are presented by the boxplots in Figs 8–11.

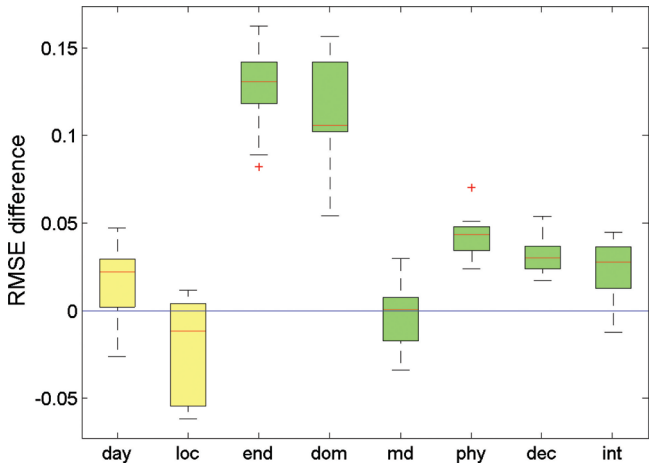

Figure 8. Performance differences between the CUB-MF model when employing the irrelevant context variable *time* and the CUB-MF model when employing each of the six relevant contextual variables (green) or each of the two unclassified contextual variables (yellow).

To visualize the difference in the performance between the baseline predictors and the CUB-MF model contextualized by each contextual variable, we calculated the differences between the RMSE values on each fold as $RMSE_{baseline} - RMSE_{context}$. The distributions of the differences between the performance of the baseline predictor MF and the CUB-MF model contextualized by each contextual variable are presented by the boxplots in Fig. 12.

The distributions of the differences between the performance of the baseline predictor AVG and the CUB-MF model contextualized by each contextual variable are presented by the boxplots in Fig. 13.

6.3. Discussion

From the results provided in the previous section, we can conclude the following.

First of all, an a-priori power analysis showed that our sample size was large enough to use the statistical tests for the detection.

From Table 3 we can conclude that the Pearson χ^2 test and the Freeman–Halton test provided consistent results for all the contextual variables. However, due to Cochran's rule, the Pearson χ^2 test could not be conducted for three variables, weather, social state and end emotion. This points to the importance of the Freeman–Halton test, which is more sample-size independent and thus suitable during the cold-start phase.

The variables *time*, *season*, *weather* and *social state* were detected as irrelevant by the statistical testing with the power analysis. *End emotion*, *dominant emotion*, *mood*, *physical state*, *decision* and *interaction* were detected as relevant. The variables *day type* and *location* could not be rejected due to the statistical

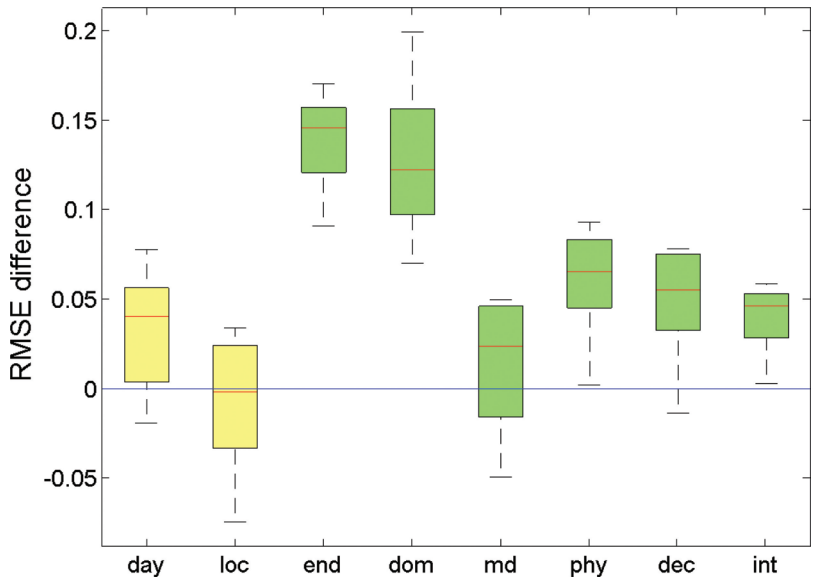

Figure 9. Performance differences between the CUB-MF model when employing the irrelevant context variable *season* and the CUB-MF model when employing each of the six relevant contextual variables (green) or each of the two unclassified contextual variables (yellow).

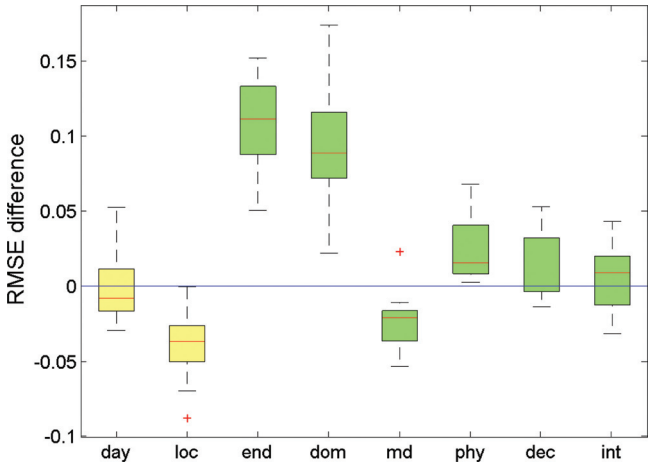

Figure 10. Performance differences between the CUB-MF model when employing the irrelevant context variable *weather* and the CUB-MF model when employing each of the six relevant contextual variables (green) or each of the two unclassified contextual variables (yellow).

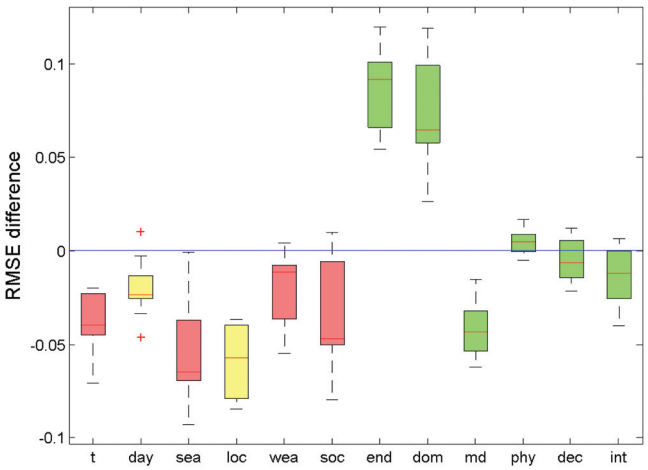

Figure 12. Performance differences between the baseline predictor MF and the CUB-MF model when employing each of the contextual variables (irrelevant in red, unclassified in yellow and relevant in green).

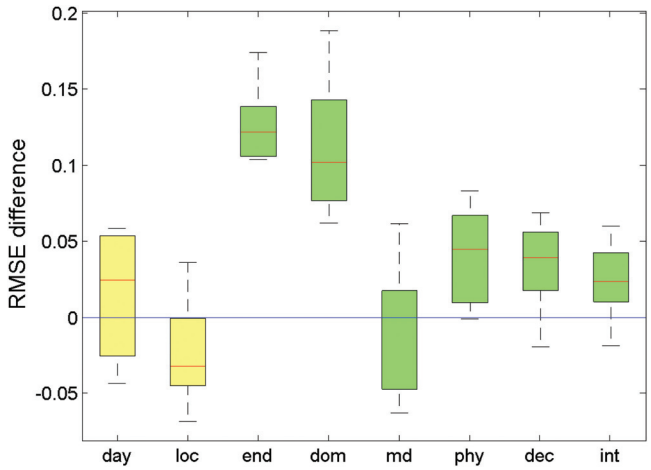

Figure 11. Performance differences between the CUB-MF model when employing the irrelevant context variable *social* and the CUB-MF model when employing each of the six relevant contextual variables (green) or each of the two unclassified contextual variables (yellow).

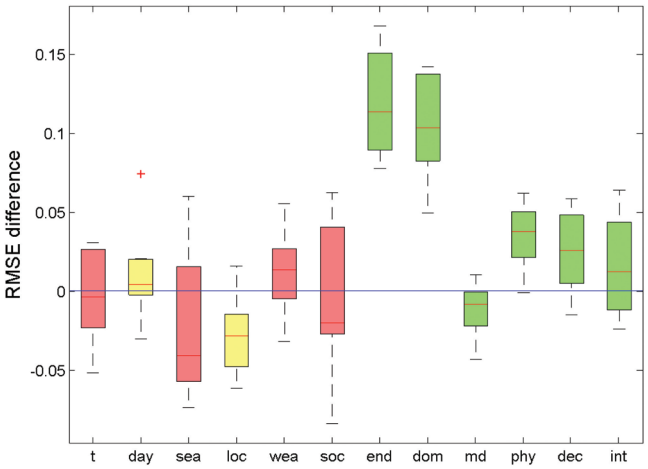

Figure 13. Performance differences between the baseline predictor AVG and the CUB-MF model when employing each of the contextual variables (irrelevant in red, unclassified in yellow and relevant in green).

power analysis explained in Section 5.2.2, and were thus labeled as unclassified variables.

As stated earlier, the CIB-MF model did not lead to conclusive results due to the small number of ratings per item. However, this is also an interesting finding since it points to the need for the inspection of the tail in the distribution of the ratings per entity whose parameter is going to be contextualized.

The CPF-MF model was also not conclusive since the feature vectors in the matrix factorization explain a smaller part of the variance in the ratings than the biases. This does not mean that the users' feature vectors should not be contextualized. It simply did not serve as a good tool for the evaluation of the relevancy detection procedure in our case.

The CUB-MF model, however, clearly shows the differences of employing different contextual variables. By inspecting the results of the Wilcoxon signed-rank test, presented in Table 4, and the boxplots in Figs 8–11, we can determine whether the model that employs the relevant contextual variable performed better than the one that employs the irrelevant contextual variable, and if the difference was significant. For example, if we inspect the first column of Table 4 and Fig. 8, we can determine that the relevant variables *end emotion*, *dominant*

emotion, *physical state*, *decision* and *interaction* all performed better than the irrelevant variable *time*, and that the difference was significant. By inspecting the second and the fourth column of Table 4 along with Figs 9 and 11, respectively, we can come to the same conclusion with respect to the irrelevant variables *season* and *social*. In Fig. 10 we can see how the relevant variables *decision* and *interaction* did not perform significantly different from the irrelevant variable *weather*. The variable *mood* was the only variable detected as relevant that performed significantly worse than an irrelevant variable, namely *weather*. It is interesting that the emotional state used as a contextual variable improved the rating prediction drastically, which is in line with the findings by Tkalčič *et al*. (2010a). To conclude, apart from the variable *mood*, we found that in overall the contextual variables detected as relevant perform better than the irrelevant ones.

By inspecting Figs 12 and 13 we can determine the impact of the relevant and the irrelevant variables on the performance, with respect to the uncontextualized baseline predictors. Figure 12 shows that the relevant variables perform equal or better (with significant difference) than the uncontextualized matrix factorization algorithm, the only anomaly, once again, being the *mood* variable. The irrelevant variables, on the other hand, perform equal or worse (with significant difference) than the baseline predictor. The same conclusion can be made for the AVG baseline predictor (see Fig. 13). The fact that for some variables there is no significant difference can be explained by the low number of ratings for a specific context value, which leads to using the uncontextualized bias, as explained in Section 5.3. To conclude, apart from the variable *mood*, we found that in overall the contextual variables detected as relevant tend to perform better than the uncontextualized models, while the contextual variables detected as irrelevant tend to perform worse than the uncontextualized models, if there are enough ratings per each context variable value during training.

For the two unclassified variables, the variable *day type* shows the tendency of improving the results (significantly in some cases), whereas the variable *location* leads degrading the results. This difference in the outcome of the model employing these variables points to the importance of the proposed usage of power analysis in the detection.

The variable *mood* was the only variable that acted differently than expected. Even though it was detected as relevant, employing this variable resulted in higher error than when employing the irrelevant variable *weather*, or the uncontextualized matrix factorization. This could be explained by the high sparsity in the *negative mood* category of the *mood* contextual variable (which is an isolated case in our dataset). There are only 89 occurrences of this context value in the dataset, which is enough to detect the association with the ratings; however, the users' contextualized biases under this condition could be under trained. This could explain the detected relevance which did not reflect in the rating prediction. The effect should however be examined further in the future work.

7. CONCLUSION AND FURTHER WORK

In this work we tackled the issues of defining and detecting contextual information in RSs. We proposed several theoretical concepts that are important for predicting which pieces of contextual information could be relevant for a specific service and how to acquire and use them. Then we created an online application for data acquisition and used these theoretical concepts to acquire a context-rich movie-RS dataset. On this dataset, we conducted statistical testing together with power analysis in order to detect the relevant contextual information that should be used for context-aware recommendations. Three different context-aware recommendation approaches, all based on matrix factorization, were implemented and tested in order to evaluate the impact of both pieces of contextual information detected as relevant and irrelevant on the ratings prediction.

The obtained results confirmed that the detection procedure is very important for (i) selecting the contextual variables that will lead to better performance of the rating prediction and (ii) avoiding the information that degrades the prediction. The power analysis also proved to be crucial for avoiding the rejection of contextual information that would in fact enhance the recommendation procedure (*day type* context). Our results also proved that when facing a small number of samples in the dataset (e.g. the cold-start phase), the Pearson χ^2 test can and should be replaced with the Freeman–Halton test.

The main drawback of our experiment is its relatively small dataset, which caused several research obstacles resulting from data sparsity. Unfortunately, we were unable to find a similar existing dataset that contained multiple, real contextual information obtained in a satisfactory way. However, the power analysis we conducted shows that our analysis is correct, and with an increasing amount of data the effect size will not change dramatically, and therefore the observations from the power analysis will become even more pronounced.

Our future work consists of continuously acquiring more data for our dataset, the searching and inspecting of existing datasets, and verifying and upgrading both the contextual information detection method and the context-aware recommendation algorithms. We will also test our methods on other applications, and make the evaluation with other measures like precision, recall and also real-world measures like users' satisfaction, novelty, serendipity, diversity etc. Furthermore, we will study the effect of the detection on using multiple context variables in different combinations, and finding the optimal combination of variables to use. We would also like to inspect the mutual dependencies of different context variables. By doing so, redundant variables could be detected and avoided. It could be that several variables convey the same or similar information so the acquisition of some of the variables could be unnecessary.

ACKNOWLEDGEMENTS

We would like to thank our colleagues Matevž Kunaver and Tomaž Pozrl from the LDOS group (http://www.ldos.si), and all our users for their help and participation in the dataset acquisition procedure.

FUNDING

This work was funded by the Slovenian Research Agency (ARRS) under grant number R-819.

REFERENCES

Adomavicius, G., Sankaranarayanan, R., Sen, S. and Tuzhilin, A. (2005) Incorporating contextual information in recommender systems using a multidimensional approach. ACM Trans. Inform. Syst., 23, 103–145.

Adomavicius, G. and Tuzhilin, A. (2005) Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng., 17, 734–749.

Adomavicius, G. and Tuzhilin, A. (2011) Context-aware recommender systems. In Ricci, F., Rokach, L., Shapira, B. and Kantor, P.B. (eds), Recommender Systems Handbook, Springer, USA, 217–253.

Agresti, A. (1992) A survey of exact inference for contingency tables. Statistic. Sci., 7, 131–153.

Baltrunas, L., Amatriain, X. and Augusta, V. (2009) Towards Time-Dependant Recommendation based on Implicit Feedback. In Proc. RecSys 2009 Workshop on Context-aware Recommender Systems, New York City, NY, USA. pp. 1–5.

Baltrunas, L., Ludwig, B., Peer, S. and Ricci, F. (2011) Context relevance assessment and exploitation in mobile recommender systems. Pers. Ubiquitous Comput., 16, 507–526.

Baltrunas, L., Ludwig, B. and Ricci, F. (2010) Matrix Factorization Techniques for Context Aware Recommendation. In Proc. Fifth ACM Conf. Recommender Systems, Chicago, Illinois, USA, pp. 301–304.

Cohen, J. (1988) Statistical Power Analysis for the Behavioral Sciences. Lawrence Erlbaum, Hillsdale, New Jersey.

Dey, A. and Abowd, G. (1999) Towards a Better Understanding of Context and Context-Awareness. In Proc. 1st Int. Symp. Handheld and Ubiquitous Computing, Karlsruhe, Germany, pp. 304–307.

Dey, A.K. (2001) Understanding and using context. Pers. Ubiquitous Comput., 5, 4–7.

Díez, F., Chavarriaga, J.E., Campos, P.G. and Bellogín, A. (2010) Movie Recommendations based in Explicit and Implicit Features Extracted from the Filmtipset Dataset. In Proc. Workshop on Context-Aware Movie Recommendation, Barcelona, Spain, pp. 45–52.

Ekman, P. and Davidson, R. (1994) The Nature of Emotion: Fundamental Questions. Oxford University Press, New York.

Faul, F., Erdfelder, E., Buchner, A. and Lang, A-G. (2009) Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses. Behav. Res. Methods, 41, 1149–1160.

Gantner, Z., Rendle, S. and Schmidt-Thieme, L. (2010) Factorization Models for Context-Time-Aware Movie Recommendations Encoding Time as Context. In Proc. Workshop on Context-Aware Movie Recommendation, Barcelona, Spain, pp. 14–19.

Gonzalez, G., de la Rosa, J.L., Montaner, M. and Delfin, S. (2007) Embedding emotional context in recommender systems. In 2007 IEEE 23rd Int. Conf. Data Engineering Workshop, IEEE Computer Society, Istanbul, Turkey, pp. 845–852.

Herlocker, J., Konstan, J., Terveen, L. and Riedl, J. (2004) Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst., 22, 5–53.

Hosseini-Pozveh, M. (2009) A multidimensional approach for context-aware recommendation in mobile commerce. J. Comput. Sci., 3, 657–663.

Joung, Y., Zarki, M.E. and Jain, R. (2009) A User Model for Personalization Services. In 2009 Fourth Int. Conf. Digital Information Management, IEEE, Ann Arbor, Michigan, USA, 1–6.

Kader, G.D. and Perry, M. (2007) Variability for categorical variables. J. Statistic. Educ., 15, 1–16.

Koren, Y. (2008) Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model. In Proc. 14th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, ACM, Las Vegas, Nevada, USA, pp. 426–434.

Koren, Y. (2010) Collaborative filtering with temporal dynamics. Commun. ACM, 53, 89–97.

Koren, Y., Bell, R. and Volinsky, C. (2009) Matrix factorization techniques for recommender systems. Computer, 42, 30–37.

Liu, L., Lecue, F., Mehandjiev, N. and Xu, L. (2010) Using Context Similarity for Service Recommendation. In 2010 IEEE Fourth Int. Conf. Semantic Computing, IEEE Computer Society, Pittsburgh, PA, USA, 277–284.

Ono, C., Takishima, Y., Motomura, Y. and Asoh, H. (2009) Context-Aware Preference Model Based on a Study of Difference Between Real and Supposed Situation Data. In User Modeling, Adaptation, and Personalization, vol 5535. Lecture Notes in Computer Science, pp. 102–113.

Park, Y. and Tuzhilin, A. (2008) The Long Tail of Recommender Systems and How to Leverage it. In Proc. 2008 ACM Conf. Recommender systems, pp. 11–18. ACM Press, Lausanne, Switzerland.

Rahmani, H., Piccart, B. and Blockeel, H. (2010) Three Complementary Approaches to Context Aware Movie Recommendation. In Proc. Workshop Context-Aware Movie Recommendation, ACM, Barcelona, Spain, pp. 57–60.

Rendle, S. (2010) Context-Aware Ranking with Factorization Models. Springer, New York.

Su, J., Yeh, H., Yu, P. and Tseng, V. (2010) Music recommendation using content and context information mining. IEEE Intell. Syst., 25, 16–26.

Tkalčič, M., Burnik, U., Košir, A. (2010a) Using affective parameters in a content-based recommender system for images. User Model. User-Adapt. Interact, 20, 279–311.

Tkalčič, M., Košir, A. and Tasič, J. (2011) Affective Recommender Systems: the Role of Emotions in Recommender Systems. In RecSys 2011 Workshop on Human Decision-Making in Recommender Systems, ACM, Chicago, IL, USA, pp. 9–13.

Tkalčič, M., Odić, A., Košir, A., Tasič, J.F. (2010b) Comparison of an Emotion Detection Technique on Posed and Spontaneous Datasets. V: ZAJC, Baldomir (ur.), TROST, Andrej (ur.). Zbornik devetnajste mednarodne Elektrotehniške in računalniške konference ERK 2010, Portorož, Slovenija, 20–22. september 2010. Ljubljana: IEEE Region 8, Slovenska sekcija IEEE, 2010, zv. A, str. 225–228, ilustr. [COBISS.SI-ID 7905364].

Toutain, F., Bouabdallah, A., Zemek, R. and Daloz, C. (2011) Interpersonal Context-Aware Communication Services. IEEE Commun. Mag., 49, 68–74.

Woerndl, W. and Groh, G. (2007) Utilizing Physical and Social Context to Improve Recommender Systems. In 2007 IEEE/WIC/ACM Int. Conf. Web Intelligence and Intelligent Agent Technology—Workshops, IEEE Computer Society, Silicon Valley, USA, 123–128.

Woerndl, W. and Schlichter, J. (2007) Introducing Context into Recommender Systems. In 2007 Workshop on Recommender Systems in e-Commerce, Vancouver, Canada, pp. 138–140.

Yap, G.E., Tan, A.H. and Pang, H.H. (2007) Discovering and exploiting causal dependencies for robust mobile context-aware recommenders. IEEE Trans. Knowl. Data Eng., 19, 977–992.

Yujie, Z. and Licai, W. (2010) Some Challenges for Context-aware Recommender Systems. In 2010 5th Int. Conf. Computer Science and Education (ICCSE), IEEE, Hefei, China, pp. 362–365.