A Short Overview of the PEOPLEVIEWS MOBILE User Interface

Angela Promitzer\textsuperscript{1} and Alexander Felfernig\textsuperscript{1} and Michael Schwarz\textsuperscript{1} and Thomas Ulz\textsuperscript{1} and Amal Shehadeh\textsuperscript{1} and Sarah Haas\textsuperscript{1}

Abstract. The PEOPLEVIEWS environment provides functionalities that support the development and application of constraint-based recommenders on the basis of the concepts of Human Computation. Domain experts are defining relationships between customer requirements and corresponding items on their own by providing answers to simple micro-tasks. On the basis of these evaluations, PEOPLEVIEWS automatically determines a set of constraints that then represent a recommendation knowledge base. Besides an HTML-5 version, PEOPLEVIEWS also includes a mobile iOS Client – a major focus of this paper is to present the corresponding user interface.

1 Introduction

PEOPLEVIEWS\textsuperscript{2} is a Human Computation \cite{2, 10} based recommendation environment that supports the definition, maintenance, and execution of constraint-based recommenders \cite{1}. The concept of PEOPLEVIEWS is to provide recommendation functionalities in a domain-independent fashion, i.e., the same environment can be applied to different item domains such as digital cameras, holiday destinations, and financial services.

PEOPLEVIEWS recommenders are based on a recommendation knowledge base \cite{3} that consists of a set of constraints automatically derived from item feedback provided by domain experts on the basis of micro-tasks \cite{6}. The underlying principle is to collect user feedback on the appropriateness of items for certain application contexts and to automatically decide \cite{4} (on the basis of the given user feedback) in which contexts which items should be recommended.

The PEOPLEVIEWS environment supports different types of micro-tasks which are simple tasks that can be completed, for example, by product domain experts without a corresponding computer science background. With the exception of one task-type all the other ones are used to acquire recommendation knowledge, i.e., in which context which items should be recommended to the user. The mentioned exception is a CAPTCHA-style micro-task which is used for the purpose of manipulation detection, for example, for identifying bots.

The major goal of this paper is to provide an overview of the PEOPLEVIEWS MOBILE user interface which is an iOS-based application that supports the answering of micro-tasks and the application of individual recommenders. We discuss the user interface on the basis of examples from the domains of mobile phones, skiing resorts, and tourist destinations (cities).

1 Graz University of Technology, Austria, email: \{a.promitzer, michael.schwarz, thomas.ulz, sarah.haas\}@student.tugraz.at, \{alexander.felfernig, ashehade\}@ist.tugraz.at.

2 Recommender Systems based on Human Computation (PEOPLEVIEWS) is a project funded by the Austrian Research Promotion Agency (843492).

The remainder of this paper is organized as follows. In Section 2 we provide an overview of different user interface elements of the PEOPLEVIEWS MOBILE iOS app. In Section 3 we provide a short overview of related work related to the application of human computation in recommender systems. In Section 4 we provide an overview of open issues for future work and conclude the paper.

2 PEOPLEVIEWS MOBILE User Interface

In this section we focus on a discussion of the PEOPLEVIEWS MOBILE user interface. When activating the environment, the first screen a user can see is the start screen depicted in Figure 1.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{peopleviews_start_screen.png}
\caption{PEOPLEVIEWS start screen (iOS version).}
\end{figure}

This screen allows a user to select a recommender application (via the list of popular recommenders or via the corresponding menu item), to enter the micro-task area in order to contribute to the development and maintenance of PEOPLEVIEWS recommenders, to play...
against the PEOPLEVIEWS community, and to adapt the setting in his/her user profile. Furthermore, the recent PEOPLEVIEWS rank is shown to the user.

If a user decides to support the development and maintenance of PEOPLEVIEWS recommenders, he/she can activate micro-tasks already assigned to the user. Micro-tasks are assigned to users on the basis of a set of selection criteria such as content-based similarity of a new micro-task with the interaction history of the user (i.e., with which recommenders and micro-tasks the user already interacted in the past). Micro-task selection is based on the dimensions of importance (how important is the completion of a micro-task?), ability (does the user have the qualification to complete a micro-task?), and interest (does the user have the time to complete a micro-task?) [8].

An example of a PEOPLEVIEWS micro-task is depicted in Figure 2. In this micro-task, the task of the user is to evaluate a skiing-resort with regard to the property quality of lift facilities. In his context, a support level can be specified, to which extent the corresponding property is fulfilled, for example, if the skiing resort has some new lift facilities but the majority is not new, the corresponding support value would be rather low.

An alternative to the micro-task interface depicted in Figure 2 are built-in PEOPLEVIEWS games. If a user interacts which such a game, he/she also contributes to the underlying recommender knowledge base. Feedback given by users is used in a similar fashion as the feedback collected by micro-tasks, i.e., is used to build a corresponding recommender knowledge base. The game interface is similar to the interface depicted in Figure 2.

Highscore lists related to the game are shown to the user after the completion of a game. An example of such a highscore list is depicted in Figure 3. Furthermore, Figure 4 depicts a game statistics that visualizes the deviation of the user answers from the “ideal” answers, i.e., answers provided on a average by other users who interacted with the game.

In Figure 4, the differences between support values estimated by other users and the support values estimated by the current user (player) are visualized by showing the distance on the corresponding rating scale.

![Game summary](image)

**Figure 3.** PEOPLEVIEWS game interface (highscore list).

When interacting with a PEOPLEVIEWS recommender application, users enter a set of preferences (answers to questions) and the corresponding recommender determines a set of items (for details see [3]). Figure 5 depicts an example list of mobile phones that have been identified as recommendations for the current user.

In the following, pairs of recommended items can be selected in order to activate an item comparison where the individual advantages and disadvantages of items get visualized. An example of such an item comparison user interface is depicted in Figure 6. In this context, the two mobile phones HUAWEI Mate 8 DS and Sony Yperia M4 Acqua are compared to each other.

The support values for individual attribute values are shown on the basis of a spider diagram which can be used to visualize the corresponding advantages and disadvantages on a graphical level. For example, one alternative in Figure 6 has in the majority been evaluated to have an excellent performance whereas the support value for very good performance is higher for the other alternative.

The view on a PEOPLEVIEWS profile is depicted in Figure 7. The profile includes information regarding username and e-mail address. Furthermore, the PEOPLEVIEWS points (PV) points of the user are shown. Information not directly shown in the user interface is related to the user’s activities in the context of recommender knowledge base development and maintenance. For example, the profile includes information related to the recommenders the user interacted with and
also provided contributions to corresponding micro-tasks. Furthermore, information is stored that reflects the quality of provided answers with regard to certain micro-task topics. This knowledge is exploited by the integrated scheduling mechanisms that determine which micro-tasks should be assigned to which users.

Finally, we want to mention that micro-tasks are internally represented in the form of RECL (recommendation language) representations. The lightweight version of RECL is used to represent micro-tasks, the full-fledged RECL version is used for the representation of PEOPLEVIEWS-internal constraint structures (also denoted as recommendation-relevant filter constraints [3]).

3 Related Work

Constraint-based recommenders [1] exploit constraints for the determination of recommendations. Such recommenders are used in situations with high-involvement items where users invest more time and efforts to assure a high-quality choice. The items to be recommended are more complex which is a major reason why constraint-based representations are needed – there often exist constraints regarding the combination of different item features and not for each combination of customer requirements a corresponding item can be found. Examples of such items are electronic equipment and financial services.

Human Computation style approaches have already been applied in the context of recommender systems. For example, WikiLens [7] is a recommendation environment that provides user communities the possibility to cooperatively develop recommenders in an open environment where users are able to introduce new items, comment on items, and search for items. The basic principles of community-maintained recommenders [7] are taken into account in PEOPLEVIEWS. Another application of Human Computation in the recommendation context is the work of [9] (Matchin).
Figure 7. PeopleViews user profile representing the individual user settings such as profile picture, e-mail address, and PeopleViews score.

Matchin is based on the idea of eliciting preferences for large datasets by asking users what a "random" person would prefer when having to choose between alternatives. The difference between the work of [7, 9] and PeopleViews is that PeopleViews-collected and -compiled recommendation knowledge can be exploited for constraint-based recommenders [1] that include functionalities such as deep explanations and intelligent diagnosis and repair for inconsistent requirements [5].

4 Conclusions and Future Work

The major focus of this paper is to provide an overview of the PeopleViews Mobile user interface. The mobile version of the system provides full access to the PeopleViews recommenders, allows the completion of micro-tasks, and also game-playing. Currently, PeopleViews is a research prototype; our major goals for future work are the following: first, we will conduct further user studies to improve system usability and empirical studies related to the prediction quality of recommendation and micro-task scheduling algorithms. Second, we will focus on the development of a mobile client for Android based systems. Finally, we will focus on extending the applicability of the PeopleViews micro-task scheduling approaches to domains such as quality assurance in requirements engineering.

REFERENCES


