

User Profiling in the Chronobot/Virtual Classroom System

Xin Li and Shi-Kuo Chang
Department of Computer Science,
University of Pittsburgh, USA,
{flying, chang}@cs.pitt.edu

Abstract

The Chronobot/Virtual Classroom (CVC) system is a novel time knowledge exchange platform where any pair of users can exchange their time and knowledge. User profile that contains user attributes, preferences, and learning patterns serves as a primary basis to identify exchange partners and determine exchange rates. In this paper, we described the methodology to acquire knowledge about users i.e. user profile from their activities. The association between user profile and user behaviors (e.g. online reading, chatting and time/knowledge exchanging) is identified by several feedback indicators extracted from browsing history, chatting session and exchange transaction. A linear learning model is constructed to fuse multiple feedback indicators to infer user preference. The methods utilizing user profile to identify the exchange partners and determine the exchange rate are also described in detail.

1. Introduction

Comparing with the traditional face-to-face style teaching and learning, e-Learning is indeed a revolutionary way to provide education in life long term. Nowadays more and more people have benefited from various e-Learning programs. However, present-day e-Learning systems are still too rigid and do not lend themselves to the peer-to-peer learning in which any users can exchange their knowledge with any others. Chronobot/Virtual Classroom (CVC) [3] is a novel time knowledge exchange platform where any pair of users can exchange their time and knowledge. The chronobot is a time manage tool for storing and borrowing time. Using chronobot one can borrow time from some one and return time to the same person or someone else. The virtual classroom is a versatile communication tool that combines the functions of web browser, chatting room, white board, and multimedia display. The CVC system is an integration of chronobot and virtual classroom that allows users freely to switch between these two applications and get maximum benefits from both.

For example, illustrated in figure 1, George, Bill, and Suzie are all students who are doing a group project together in a graphics design course in which they use the CVC system to collaborate with each other. The whole

project is divided into several tasks, each of which is mainly assigned to one person. George meets a problem in his task and can not solve it by himself. So he interacts with Bill and Suzie in the virtual classroom, and eventually they help him out. However, in order to keep workload even among teammates, George has to put in efforts either in the past or in the future to help Bill and Suzie. The chronobot serves as a platform for them to do such time and knowledge exchanges which could have significant value for many applications. Many more interesting scenarios can be found in [3].

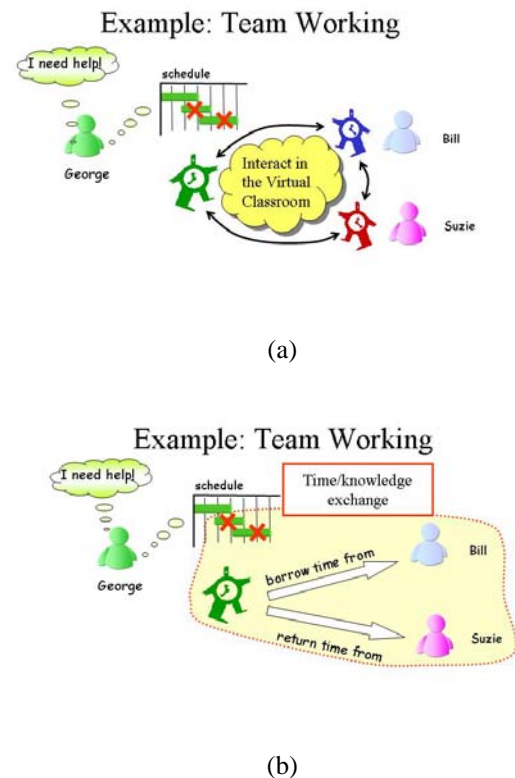


Figure 1. Application of the CVC system:
(a) Communication in Virtual Classroom; (b) Time and knowledge exchange in Chronobot

Generally speaking, a transaction of the time/knowledge exchange in the CVC system includes the following steps:

1. Identify a slice of time or knowledge for exchange;
2. Search for exchange partner or partners;
3. Perform time or knowledge exchange through bidding and negotiation;
4. Manage the exchange slice of time or knowledge;

5. Provide feedback on the results.

In these processes, user profile that contains user attributes, preferences and learning patterns plays a vital role because:

- It provides fundamental information to identify time or knowledge for exchange and exchange partners.
- It serves as a primary basis to determine time/knowledge exchange rate.

In this paper, we describe an approach to acquire knowledge about users represented by user profile based upon user activities such as web browsing, chatting and time/knowledge exchanging in the CVC system. We believe that our work has significant value, because the acquisition of knowledge about users is a key process not only in our system but also in many other applications. For example, the recommendation systems [1, 4, 6, 9] mainly depend on user profiles in terms of similarity and differences to provide particular suggestions. The personalized web search engine [10] can construct user profiles from browsing history and consequently provide personalized results to match the information needs of individuals. Compared to these applications, an effective user profiling is much more feasible in our system because of the following two reasons:

1. Time/knowledge exchange (i.e. peer-to-peer learning) is a much more continuous process than the activities (e.g. online news reading and web searching) in many other applications.
2. Multiple data sources can be employed to assess user profile in our system. For example, in addition to browsing history, chatting session and knowledge/time exchange transaction can also serve as important input sources for the profiling process.

User preference is key information in user profile. As far as we studied, the majority of user profiling approaches mainly depends on the user feedbacks to retrieve user preference. The feedback can be assessed explicitly by rating, or implicitly by the user behaviors such as print and save. In this paper, we are not advocating either of these two approaches because both of them have significant advantage and disadvantages [10]. Instead we propose a methodology which can combine multiple feedback measures to get more complete and accurate assessment. User preference can be inferred on the basis of data from three sources, i.e. browsing history, chatting session, and knowledge/time exchange transaction. A linear learning model is constructed to fuse all the related data for the inference of user preference. Five feedback indicators – reading time, scroll number and print/save from browsing history, relational index from chatting session, and the exchange index from knowledge/time exchange transaction serve as input variables of the model. Demonstrated by the experiments in the prototype system, the proposed model can infer user preference much more accurately than any single one of these indicators. The applications of user profile – to identify

the exchange partners and determine the exchange rate are also described in detail. A preliminary report about this work has been published in [7].

The rest of this paper is organized as follows: two basic concepts *user profile* and *ontology knowledge base* in our system are described in section 2 and 3 respectively. Section 4 discusses the measures to identify the association between user activities and preferences, where five implicit feedback indicators are defined. The learning model to fuse these indicators for a final assessment of user preference is described in Section 5. The application of user profile is described in Section 6. In Section 7, our prototype system and experiments in it are described. The related research is discussed in Section 8, followed by brief conclusions in Section 9.

2. User Profile

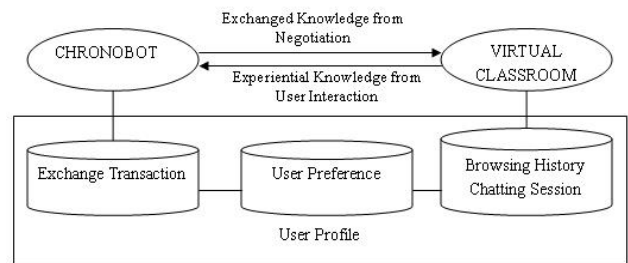


Figure 2 User Profile in the CVC System

The *user profile* is the physical realization of the *user model*, which is an abstraction of the user preferences and characteristics. Shown in figure 2, in the CVC system a user profile *upf* is organized as a 6-tuple:

$$upf = \langle id, user-attributes, browsing-history, chatting-session, exchange-transaction, preference \rangle$$

- *id* is a unique identification number.
- *user-attributes* is a vector $A(u) = (a_1(u), a_2(u), \dots, a_n(u)) = (x_1, x_2, \dots, x_n)$, where $a_i(u) = x_i$ is the i^{th} attribute for the user u – it could be user’s name, expertise level, schedule and so on.
- *browsing-history* is a set of the pages which user has visited. For the purpose of user profiling, the corresponding behaviors are recorded for each page, e.g. reading time, number of scrolls and print/save.
- *chatting-session* is a set of conversations which user participated in the virtual classroom. All the contents are recorded in natural language.
- *exchange-transaction* is a set of time/knowledge exchange transactions which user performed in the chronobot.
- *preference* is a vector $P(u) = (p_1(u), p_2(u), \dots, p_n(u)) = (y_1, y_2, \dots, y_n)$, where $p_i(u) = y_i$ is the preference of user on the i^{th} topic. The topics are defined by the *ontology knowledge base* (which will be discussed in the next section).

When a user first registers the CVC system, the user is asked to enter information such as personal data, areas of expertise, levels on these areas and so on. The *user*

profile manager provides an HTML front-end using which new users can register themselves with the system. During the registration, the *user profile manager* collects important information from the users such as the user id, name, address, credit card details, areas of experience, levels, skill set, the hourly rate, e-mail address and so on. The rationale behind having the credit card information is that if the user defaults in time/knowledge exchange transactions, then his/her credit card is billed depending upon the number of hours defaulted.

User preference is the most valuable information in user profile. However, it is usually hard to be assessed directly, because in many cases it is difficult to request users to express their interests explicitly -- it is simply too much work for them. Furthermore, users may change their preferences upon time, and hence they can not be assessed statically. For this reason, in our approach user preference is not directly input by users, but implicitly inferred from user behaviors.

3. Resource Organization

The learning resources in the CVC system are all web-based multimedia materials, which are organized by the ontology knowledge base (OKB). All the topics and their relations (i.e. mainly *subtopic* relation) are described in the OKB. For example, a small part of the OKB in our system is shown in Figure 3.

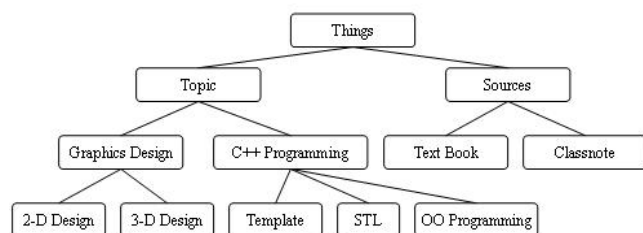


Figure 3: A small part of OKB tree

Based upon the OKB, a learning resource lr in the CVC system is defined as a tuple:

$$lr = \langle url, topic, keywords \rangle$$

where

- url is the universal resource locator which is an identification string of lr .
- $topic$ is a concept in the OKB which is the subject of lr .
- $keywords$ is the set of key words of lr . They are closely related to the content of lr , but usually not in the OKB.

In practice, in order to build an efficient connection between the resources and user profile, two kinds of mapping are employed in our system:

- One-to-many mapping from topics to URLs.
- One-to-many mapping from topics to keywords.

Topics	Keywords	URLs
Graphics Design	Layout, Rendering, ...	http://www.cs.pitt.edu/flying/CVC/graphics/*
C++ Programming	Template, STL, ...	http://www.cs.pitt.edu/flying/CVC/cplusplus/*

Figure 4. An example of topics to URLs and keywords mappings

Figure 4 shows examples of these two kinds of mapping. The star "*" represents any combination of characters in URLs. The rationale behind these mappings is that they can greatly facilitate the aggregation of feedback indicators on topics.

4. Feedback Extraction

There are three user feedback sources in our system: browsing history, chatting session, and time/knowledge exchange transaction. In this section, we discuss the methods to extract the knowledge about users, i.e. user preference, from these three sources respectively.

Totally five distinct feedback indicators are defined. In fact, we can design these indicators to be as complex as is necessary for the intended application. However for practicality it is important to keep them simple and manageable.

4.1 Browsing History

There is no doubt that browsing history conveys significant information for inferring user preference. However, it is hard to determine interests of users just based on the pages they visited, because it always happens that users open a page they don't like or just by mistake. Aiming at the more accurate assessment, three feedback indicators are defined based upon browsing history:

1. Reading Time

Usually if users spend longer time on reading about a topic, it means that they have more interest on it. For this reason, we record reading time of the user u as a vector $RT(u) = \langle \langle lr_1, rt_1 \rangle, \langle lr_2, rt_2 \rangle, \dots, \langle lr_n, rt_n \rangle \rangle$, lr_i is the learning resource user u has visited, rt_i is the corresponding reading time.

2. Number of Scrolls

Definitely the scroll either by mouse or PageDown/PageUp key on a page is a signal of interest. For this reason, we record the number of scrolls for user u as vector $SC(u) = \langle \langle lr_1, sc_1 \rangle, \langle lr_2, sc_2 \rangle, \dots, \langle lr_n, sc_n \rangle \rangle$, lr_i is the learning resource user u has visited, sc_i is the corresponding number of scrolls.

3. Print/Save

In most cases, printing or saving a page is a strong signal of interest. For this reason, we record the set $PS(u) = \langle \langle lr_1, ps_1 \rangle, \langle lr_2, ps_2 \rangle, \dots, \langle lr_n, ps_n \rangle \rangle$, lr_i is the resource which user u has visited, ps_i is 1 if it has been printed/saved, 0 otherwise.

4.2 Chatting Session

The experiences accumulated in the virtual classroom are among the most valuable assets for preference inference. In practice the experiences are the stored transcripts of the virtual classroom sessions. These transcripts are represented as XML documents. In fact we consider everything that is exchanged or recorded in the chronobot/virtual classroom system as some form of XML document.

The Relational Index *RI* is built to support easy access of the accumulated learning experiences. The session transcripts (XML documents) are stored in an experience-base. The Relational Index is then constructed. It relates learning experiences to user preferences in the user profile. For example, if x_1, \dots, x_n are keywords specified in the topic keywords mapping discussed in section 3, the Relational Index can be used to find u_j , the user most closely related to the specified topics.

We can also use the Relational Index *RI* to relate users to keywords and/or users to users. In other words, *RI* is used to form an *association* in the information exchange process among users.

The *RI* is updated each time a new session transcript is created. The transcript is analyzed with respect to a set of pre-specified keywords x_1, \dots, x_n in the topic keywords mapping. If a dialog of user u_j in the transcript involves a keyword x_k , we can store a new record $[x_k; u_j; p]$ in *RI* where the frequency p is set to 1, or update p if such a record already exists. Similarly if a dialog between two users u_j and u_i in the transcript involves a keyword x_k , we can store a new record $[x_k; u_j; u_i; p]$ in *RI* where the frequency p is set to 1, or update p if such a record already exists.

For example the transcript is as follows:

George: Do you think we need to add 3D graphics to the presentation?

Suzie: No, I don't think so. But the layout can be improved.

George: That is good, because I still cannot find a person to do 3D graphics.

The pre-specified keyword set is:

```
{layout, graphics, 3D graphics}
```

The Relational Index, after the processing of the above transcript, contains the following records as well as other previously entered records:

```
[3D graphics; George; 2]
[3D graphics; George; Suzie; 1]
[layout; Suzie; 1]
[layout; George; Suzie; 1]
```

4.3 Exchange Transaction

The transactions of time/knowledge exchange can also be a significant indicator of user preference. In our system, the exchange of time/knowledge is implemented as a bidding process: a person who needs help from others can start a bid, providing the task description, the required knowledge, and the amount of time needed. Anyone else can place a bid to offer his/her time. A successful exchange/bid transaction includes at least the following information: bid starter, bid winner, task description, keywords, and time amount. All these information is stored in XML file in practice.

The Exchange Index *EI* is built to easy access of the accumulated exchange history. Similar as Relational Index, the *EI* is used to relate user to keywords and/or user to users. It is updated every time a new transaction is created. The task description is analyzed with the respect to a set of pre-specified keywords x_1, \dots, x_n in the topic keywords mapping. If a transaction of user u_j is related to a keyword x_k , we can store a new record $[x_k; u_j; t]$ in *EI* where time t is set to the time amount of the transaction, or update t by adding the amount if such a record already exists. The time amount is positive if user u_j borrows the time to others, otherwise it is negative. Similarly if a transaction between two users u_j and u_i involves a keyword x_k , we can store a new record $[x_k; u_j; u_i; t]$ in *EI* which means user u_j has borrowed t amount of time from u_i .

For example, George starts a bid as follows:

```
Bid Task: Help to improve the layout in a 3D graphics
design;
Time Amount: 8 hours;
```

The pre-specified keyword set is the same with the example in section 4.2.

Through a bidding process, Bill can offer 5 hours, and the rest 3 hours help can be done by Suzie. The Exchange Index, after processing these transactions, contains the following records as well as other previously entered records:

```
[3D graphics; George; -8]
[3D graphics; Bill; 5]
[3D graphics; Suzie; 3]
[3D graphics; George; Bill; 5]
[3D graphics; George; Suzie; 3]

[layout; George; -8]
[layout; George; 5]
.....
```

The Exchange Index may contain records relating multiple (more than two) users or multiple (more than one) keywords as well as the Relational Index.

5. Preference Assessment

As discussed in section 2, the preference of user u can be expressed as a vector $P(u) = (p_1(u), p_2(u), \dots, p_n(u)) = (y_1, y_2, \dots, y_n)$, where $p_i(u) = y_i$ is the preference of user u on the i^{th} topic. The topics are described in the OKB. Give a user u and a topic t , the preference of u on t can be inferred by the feedback indicators which are described in section 4. Using the mappings from topics to URLs and keywords, the five feedback indicators can be easily collected for each topic. Single topic could have multiple learning resources, and hence it could have many feedback indicators. For this reason, the following five variables are defined for any pair of u and t by aggregation and normalization:

1. rt – the average reading time per 1000 words;
2. sc – the average number of scroll per 1000 words;
3. ps – the average number of print/save per view.
4. ri – the average relational index per 1000 words in chatting sessions.
5. ei – the average exchange index per 100 hours in time knowledge exchanges.

A linear model, which can predict user preference based on feedback variables, is constructed using linear regression. In this model, the five variables mentioned above serve as the input variables, the output variable -- user preference up can be assessed by a linear combination of the input variables:

$$up = \alpha_1 \cdot rt + \alpha_2 \cdot sc + \alpha_3 \cdot ps + \alpha_4 \cdot ri + \alpha_5 \cdot ei + \beta + e \quad (1)$$

in which:

- $\alpha_i (i = 1, 2, \dots, 5)$ and β are coefficient variables, which can be assessed through linear regression on training data.
- e is the residual with the mean zero for all the training data.

For more information about linear prediction and regression, please refer to [5].

6. Application of User Profile

In our system, user profile can be applied to identify appropriate partners and determine exchange rate in time/knowledge exchange transactions. Meanwhile, user profile provides valuable information about ontology knowledge, which can possibly help to construct and update the ontology knowledge base.

6.1 Identification of Exchange Partners

The requirements for exchange partners are described in the exchange task. However, frequently there are so many candidates who are qualified. It is very important to choose a proper one who is most likely willing to perform the exchange. Assuming that users sharing more interests

on the topics have more chances to do the time knowledge exchange, a similarity function S is defined on users. Given an exchange task t , users u and v are characterized by task related preferences (x_1, \dots, x_n) , and (y_1, \dots, y_n) from their profiles. Denote the corresponding average preferences are $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n$, the similarity function S on u and v is defined using Pearson correlation coefficient:

$$S(v, u) = \frac{\sum_{i=1}^n (x_i - \bar{x}_i) \times (y_i - \bar{y}_i)}{\sqrt{\sum_{i=1}^n (x_i - \bar{x}_i)^2 \times \sum_{i=1}^n (y_i - \bar{y}_i)^2}} \quad (2)$$

A list of exchange candidates can be sorted using the similarity function S . The one with the highest similarity is always the first candidate to be approached for the exchange.

6.2 Determination of Exchange Rates

User profile provides the fundamental information to determine the exchange rates between two users. Give an exchange task t , users u and v are characterized by task related attributes (x_1, \dots, x_n) , and (y_1, \dots, y_n) from their profiles. The attributes could be the preferences, expertise levels and so on. For the two corresponding attributes x_i and y_i , the information distance measure is denoted by $d_i(x_i, y_i)$, where d_i is between 0 and 1 (a metric).

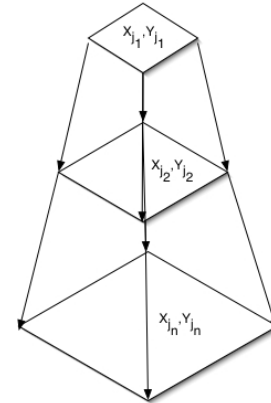


Figure 5. Determination of exchange rate.

The exchange rate between users u and v , is denoted as follows:

$$Exchange(u, v) = e^{1/\sum C_{ji} * d(x_{ji}, y_{ji})} \quad (3)$$

where the summation is over all the terms $C_{ji} * d(x_{ji}, y_{ji})$, and C_{ji} is a scaling constant.

We now illustrate the concept by presenting an example. Let us assume that George and Bill are both media artists and the two agents' primary skill matches. Therefore $C_1 d_1(x_1, y_1) = 0$. If the primary skill does not match, $C_1 d_1(x_1, y_1)$ becomes a big number. For instance, C_1 is 10,000 and d_1 is between 0 and 1, in this case close to 1. Then $C_1 d_1(x_1, y_1)$ is close to 100,000 and the exchange rate is close to 1. No need to continue.

The two agents' familiarity with subject area also is comparable, so $c_2 d_2(x_2, y_2) = 0$. If the familiarity does not match, then $C_2 d_2(x_2, y_2)$ becomes a big number. For instance, C_2 is 1,000 and d_2 is between 0 and 1, in this case close to 1. Then $C_2 d_2(x_2, y_2)$ is close to 1,000 and the exchange rate is close to 1. No need to continue either.

Finally, the two agents differ in secondary skill. Therefore $C_3 d_3(x_3, y_3)$ is small and we have an exchange rate that reflects the difference in the two agents' secondary skill. Notice the index function takes care of the re-arrangement of the relative importance of the n attributes. The constants C_j are also important. They take care of the relative scaling of the various attributes. Figure 10 illustrates the determination of the exchange rate as a dynamic process of comparing different attributes to identify the ones that really matter. Once such attributes are identified, the users can negotiate to determine the exchange rate.

6.3 Information Exchange with the OKB

Ontology knowledge plays an important role in the assessment of user profile. At the same time, the user profile retrieved from user experiences is among the most valuable assets which describe the ontology. It is always possible to construct and update ontology based upon user profile. We believe that the information exchange between user profile and ontology knowledge has significant importance, because for a long time the construction of ontology knowledge remains a difficult problem [8]. Particularly, it could be extremely cumbersome for end users to construct and maintain an ontology knowledge base by themselves. However, it is much easier for them to implicitly manipulate ontology through the time/knowledge exchanges. Here we present some preliminary works along this line by the following examples; more works in this direction will be the subject of our future research.

For example, several web pages about “*Standard Template Library*” (STL), which is a subtopic of “*C++ programming*”, are newly added into our system. In order to provide a guideline for information delivery on this new topic, we can update the ontology knowledge base by adding “*Standard Template Library*” as a subtopic of “*C++ programming*”. As a result, the users who have evident preferences on the topic “*C++ programming*” probably will be suggested for it. Alternatively, supposing that there is no efficient way to manipulate ontology, this kind of update with the new topic in ontology knowledge base can be automatically performed based upon user activities. Assuming that there are a large number of users who have visited the new web pages on “*Standard Template Library*”, their most common preference can be identified from their user profiles, which is “*C++ programming*” in this case. By knowing this, the system can infer that “*Standard Template Library*” is a subtopic

of “*C++ programming*” and update the ontology knowledge base automatically.

Obviously, this kind of inferences is not always correct. However, with the accumulation of user experiences, user profile retrieved from them will be more and more accurate, and hence the system will eventually be able to construct reasonable ontology knowledge through this way.

7. Prototype System and Experiments

A prototype CVC system has been implemented and used in real applications. Verified by the experiments in the prototype system, the proposed model can infer user preference much more accurately than any single one of these indicators.

7.1 Prototype System

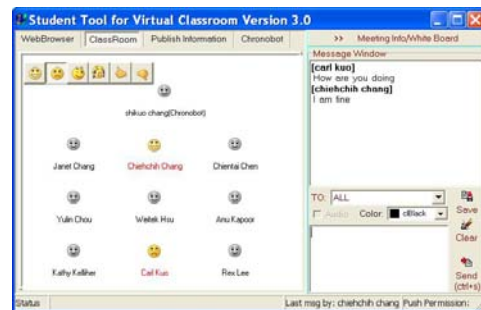


Figure 6. An example of virtual classroom

The prototype of virtual classroom system is illustrated in figure 6. Basically it is a universal communication tool. The users (students and/or instructors) represented by emotive icons can join a virtual classroom to exchange information including text messages, web pages, sketches, and audio/video clips. The system will automatically record user activities (e.g. browsing and chatting) and related information (e.g. reading time, scroll, etc.).

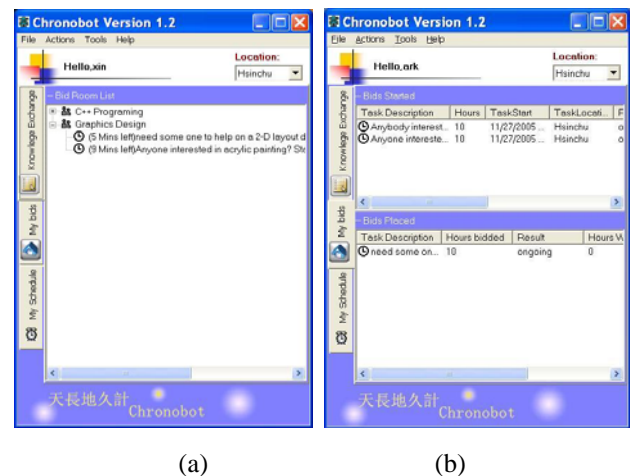


Figure 7. An example of chronobot, (a): bidding room for time knowledge exchange; (b): transaction records.

The prototype of chronobot is illustrated in figure 7.

Basically, it is a tool to exchange time and knowledge through bidding processes. Users can start new bids in different bidding rooms (figure 7(a)). A simple example of such bids is “I need some one help me on 2-D layout design for 5 hours”. Users can respond these requests by placing their own bids. For example, a user can respond by “I can contribute 2 hours on it”. The details about these transactions are recorded in user profiles (figure 7(b)).

7.2 Experiments and Result Analysis

The prototype systems are used by the students from several classes in University of Pittsburgh. The following experiment has been done to obtain the training data for our linear learning model:

- Step 1: Encourage students to use the prototype system as much as they can
- Step 2: Select a set of students who used the systems for more than a reasonable time (i.e. 72 hours). Asked them to seriously indicate their interests from 0 (least) to 5 (most) on every topic in which they have involved.
- Step 3: Input the preference indicated by the students with the corresponding feedback variables into the linear model (1), assess the coefficient variables using linear regressions.
- Step 4: Separate the testing and training data sets, verify the linear model by the cross validation.

UserID	Topic ID	Preference	<i>rt(sec.)</i>	<i>sc</i>	<i>ps</i>	<i>ri</i>	<i>ei</i>
006	073	3	65	0.2	0.07	2.6	5.4
017	135	1	23	0.1	0	1.2	0
017	009	5	107	0.1	0.34	5.3	0
...

Figure 8. Examples of experimental results

The data of the experiment are illustrated in figure 8. For each pair of user and topic, there are five feedback variables (described in Section 5) and preference. Input by these variables, the learning model in (1) can be constructed using least-squares linear regression [5].

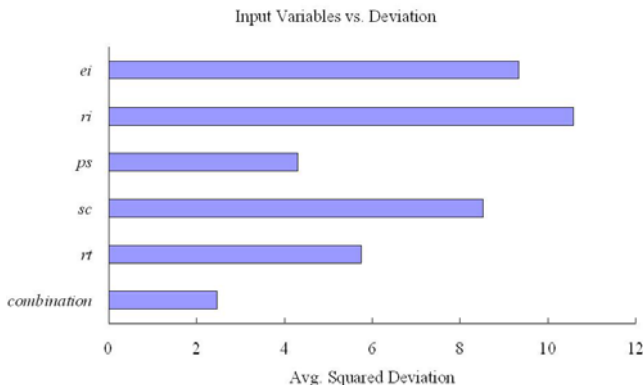


Figure 9. Input variables vs. Deviation

Using the cross validation, the model can be evaluated by the average squared deviation between the preference inferred by the linear model and the one that users explicitly indicated. Illustrated in figure 9, the

combination of multiple feedbacks is compared with single feedback: the bar labeled with “combination” represents the result input by all five feedback indicators, while other bars are the results with only one indicator as the input variable i.e. *rt*, *sc*, *ps*, *ri*, and *ei* respectively. Shown in figure 9, the combination of the indicators can assess much more accurate preference than any single indicators.

8. Related Research and Discussion

Because the time and knowledge exchange is indeed a new concept, we can hardly find similar approaches with our CVC system up to the time of this paper. However, there are numerous user profiling approaches in other application areas. As the related research, recommendation system is one of the most important applications, where user profiling is the key process collecting user feedback for items in a given domain and assessing user preference in terms of similarities and differences to determine what to recommend. Depending on underline technique, recommendation systems can be divided into collaborative filtering-based [6] content-based [4] and hybrid [1, 8] approaches. Classified by means to acquire feedback, they can be categorized as explicit rating [1, 6, 8], implicit rating [6] and no rating needed [4] systems.

In fact, user's feedbacks are so important that only very few content-based recommendation systems require neither explicit rating nor implicit rating. For example, SurfLen [4] is a recommendation system using data mining techniques to assess the association rules on web pages through user's browsing history without the feedbacks. However, it is hard to find exact interests of users just based on the browsing history, since it always happens that users open a page they don't like or just by mistake. This problem becomes even more severe in the situation that the system is sparsely used.

GroupLens system [6], which filters Usenet news, is a collaborative filtering system using n-nearest neighbor-based algorithm. In this algorithm, user profile is assessed based on a subset of appropriate n users similar to this user. The early version of GroupLens gathers user's feedback only by explicit rate. However, observing the extra costs of the explicit rating, in the latest version it also uses reading time as an implicit indicator.

Fab system [1] is also using the collaborative filtering model, meanwhile introducing the content analysis by a “topic” filtering. Web pages are initially ranked by the topic filter and then sent to user's personal filters. Users are required to give an explicit rate, and this feedback is used to modify both the personal filter and the original topic filter.

Compared to the approaches above, our approach can be classified as a hybrid approach. We assess the similarity

of users in the exact same way as the collaborative filter-based systems (shown in equation (2)). Meanwhile, the mappings from topics to keywords and URLs based on the OKB serve as simple and effective content-based filters. Particularly, our approach proposes a novel way to infer the user preference: multiple implicit feedback indicators are employed at the same time, and the final assessment are retrieved using a linear learning model. As a result, more accurate preference can be assessed without introducing any extra costs for users.

9. Conclusions

The CVC system is a novel time knowledge exchange platform that supports peer-to-peer learning. In this paper, we describe an approach to acquire and utilize the knowledge about users i.e. user profile in the CVC system. User preference – the most important information in user profile, is inferred from multiple data sources i.e. browsing history, chatting session, and time knowledge exchange transaction. The methods utilizing user profile to identify the exchange partners and determine the exchange rate are also described in detail.

10. Acknowledgements

This research is supported in part by the Industry Technology Research Institute (ITRI) and the Institute for Information Industry (III) of Taiwan. We would like to thank Chieh-Chih Chang and Jui-Hsin Huang for the valuable discussion and comments.

References

- [1] M. Balabanovi and Y. Shoham. Fab: content-based, collaborative recommendation. *Communication of the ACM*, 40(3):66–72, 1997.
- [2] M. Blochl, H. Rumershofer, and W. Wob. Individualized e-Learning systems enabled by a semantically determined adaptation of learning fragments. In *Proceeding of the 14th international workshop on Database and Expert Systems Applications*, pp. 640–645, 2003.
- [3] S. K. Chang. A chronobot for time and knowledge exchange. In *Proceeding of the 17th International Conference on Software Engineering and Knowledge Engineering*, pp. 3–10, Taipei, Taiwan, Jul. 2005.
- [4] X. Fu, J. Budzik, and K. J. Hammond. Mining navigation history for recommendation. In *Proceeding of the 5th international conference on Intelligent user interfaces*, pp. 106–112, 2000.
- [5] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer, New York, 2001.
- [6] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. GroupLens: applying collaborative filtering to usenet news. *Communication of the ACM*, 40(3):77–87, 1997.
- [7] X. Li and S. K. Chang. A Personalized E-Learning System Based on User Profile Constructed Using Information Fusion. In *Proceeding of the 11th International Conference on Distributed Multimedia Systems*. pp. 109-114, Banff, Canada, Sep. 2005.
- [8] Deborah L. McGuinness. Ontologies come of age. In Dieter Fensel, Jim Hendler, Henry Lieberman, and Wolfgang Wahlster, editors, *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. MIT Press, Boston, NY, 2002.
- [9] S. E. Middleton, N. R. Shadbolt, and D. C. D. Roure. Ontological user profiling in recommender systems. *ACM Trans. Inf. Syst.*, 22(1):54–88, 2004.
- [10] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of the 13th international conference on World Wide Web*, pages 675–684, 2004.