Plan Recognition as an Aid in Virtual Worlds

Ting Yu
Institute for Information and Communication Technologies
Faculty of Information Technology, University of Technology, Sydney,
PO Box 123, Broadway, NSW 2007, Australia
yuting@it.uts.edu.au

Simeon J. Simoff
Institute for Information and Communication Technologies
Faculty of Information Technology, University of Technology, Sydney,
PO Box 123, Broadway, NSW 2007, Australia
simeon@it.uts.edu.au

ABSTRACT
While the development of virtual worlds provides players flexible, free spaces and attractive user-interfaces, the collaboration between computers and players has not become more efficient. Developments in human-computer collaboration may come from more efficient understanding between computers and humans. This paper presents a method to use plan recognition as an aid to improve the human-computer collaboration in a virtual world. An often-used data mining method, association rules mining, is employed to create a user behavior model, which represents a player's basic characteristics. Based on this model and the current observations from players, a plan recognition algorithm tries to predict players' action sequence. By using these predictions, the virtual world can organize its local resource more efficiently and react to players' requests timelier. Moreover, some main problems in probability-based plan recognition are discussed.

Keywords
Plan recognition, Association rules, Virtual world

1. INTRODUCTION
While the development of agent-based virtual worlds provides players flexible, free spaces and attractive user-interfaces, the collaboration between computers and players has not been more efficient. Moreover, with the increase of players' amount, many problems have emerged, which include the user-interface complexity of the virtual world, the inter-agent cooperation and competition etc., so that players are often confused by some trivial, routine tasks. In the University of Technology, Sydney, a developing e-Market system is also handling these problems. The e-Market system is a multi-agent market system, provides players a virtual place to exchange their information and goods. In this system, all of players behave in a virtual world and each player is represented by a group of agents. In these agents (see Figure 1), a negotiation agent (Negotiator) retrieves and exchanges a large amount of information with opponents to achieve an agreement, and an assistant agent is developed to undertake partial functions of the negotiator. This assistant handles routine negotiation support problem and thus allows a negotiator to deal with more complex issues. The other agents retrieve information from the virtual world or even the outside world, and deliver them to the assistant agent for combination.

Being similar to e-Markets, many other large-scale complicated applications of agent-based virtual worlds are engaged in improving its human-computer collaboration. Especially some applications in the interactive entertainment, e.g. computer games, require the timely human-computer collaboration and the fluent cooperation between players, who are represented by some agents in a virtual world.

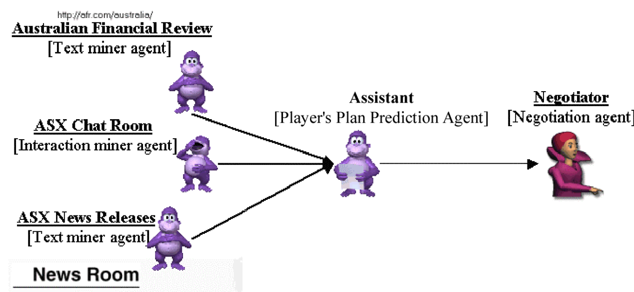

Figure 1. Negotiation support in e-markets

In this paper, only one of the tasks undertaken by the assistant is discussed: the usage of plan recognition as an aid to improve agent's ability of real-time information retrieval and cooperation with the users. The general area of inferring the goals and intentions is commonly known as plan recognition [1]. The complete plan recognition problem is extremely difficult, but even a predicting partial action sequence can help to optimize the system resource management and negotiator's information retrieval. The later converts the information pull into a kind of information push, that is, the assistant is constantly providing information and services toward negotiator, rather than negotiators taking initiative to pull some information [1].

2. ALGORITHMS
This section presents a solution of plan recognition, which consists of two main algorithms: a building algorithm of user behavior model and a plan retrieval algorithm. Our domain of interest is agent-agent interaction in a large, on-going, and dynamic environment, similar to the above discussed e-Market system. Some difficult features in this domain include the loose temporal ordering of actions, the interleaving of multiple tasks, the large space of possible plans, some sequences of actions that are shared, suspended, and lead to multiple goals [1]. The solution, presented in this paper addresses some of these complexities. The limitations of the solution are discussed in Section 5.

The solution is an interactive process: through analyzing a negotiator's past behaviors, especially the behavior frequency, its individual user behavior model is created; based on what is currently observed and using individual user behavior model, the solution assesses the various hypotheses, and selects the one with highest probability.

2.1 The Model Building Algorithm

The basic algorithm to build a user behavior model includes the transformation of action sequence data into transaction data and association rule mining. The description is begun by a straightforward formalization of action, transaction and association rule. To achieve a goal, a player has to take a series of continuous *actions*, which is treated as a *transaction*. The actions a player has done can be represented as a set of actions (an *actionset*): A = $\{A_1, A_2, A_3, \ldots, A_m\}$, therefore, a single transaction is represented as a subset of the action set, e.g. $\{A_1, A_3, A_4, A_6\}$. Further, the player's transactions consist of a transaction set: T = $\{T_1, T_2, \ldots, T_n\}$, where $T_2 \subseteq A$. Because the actions normally follow an order, different from the familiar market basket analysis, the time sequence of actions is another important criteria, which also determines the interrelationship between two elements, e.g. T2 = $\{A_1, A_3, A_4, A_6\}$ can be represented as $A_1 \rightarrow A_3 \rightarrow A_4 \rightarrow A_6$.

In the association rule mining algorithm, the *support* for an association rule $X \rightarrow Y$, e.g. $\{A_1, A_3\}$, is the percentage of transactions in the transaction set, e.g. T, that contain $X \wedge Y$, e.g. $A_1 \wedge A_3$. The *confidence* for an association rule $X \rightarrow Y$, e.g. $\{A_1, A_3\}$, is the ratio of the number of transactions that contain $X \wedge Y$, e.g. $A_1 \wedge A_3$ to the number of transactions that contain X, e.g. A_1 [9].

Suppose that a player has done a set of transactions T = {T1, T2, …, T9} (see Figure 2(A)) by using Apriori algorithm [2], it is easy for system to get the *frequent actionsets* (see Figure 2(B)) and corresponding confidence list for each frequent actionsets (see Figure 2(C)) where the minimum support is set as 2/9 and no minimum confidence is set. Here, the minimum support is used to filter noise to build a non-trivial model [7]. Considering the time sequence of actions, the system only lists the possible confidences, e.g. $A_1 \rightarrow A_2 \wedge A_5 \wedge A_7$ instead of $A_5 \wedge A_7 \rightarrow A_1 \wedge A_2$.

Transaction	Actionset
T1	$A_1 \rightarrow A_2 \rightarrow A_5 \rightarrow A_7$
T2	$A_1 \rightarrow A_2 \rightarrow A_4 \rightarrow A_6$
T3	$A_1 \rightarrow A_2 \rightarrow A_3 \rightarrow A_5 \rightarrow A_7$
T4	$A_1 \rightarrow A_2 \rightarrow A_3 \rightarrow A_4 \rightarrow A_6$
T5	$A_1 \rightarrow A_4 \rightarrow A_6$
T6	$A_2 \rightarrow A_3$
T7	$A_4 \rightarrow A_2 \rightarrow A_7$
T8	$A_1 \rightarrow A_2 \rightarrow A_5 \rightarrow A_4 \rightarrow A_6$
T9	$A_1 \rightarrow A_2 \rightarrow A_3$

(A)

Actionset	Support count
$\{A_1, A_2, A_5, A_7\}$	2/9
$\{A_1, A_2, A_4, A_6\}$	3/9
$\{A_1, A_2, A_3\}$	3/9

(B)

$A_1 \wedge A_2 \rightarrow A_5 \wedge A_7$ confidence 3/6

$A_1 \wedge A_2 \wedge A_5 \rightarrow A_7$ confidence 3/3

$A_1 \wedge A_2 \rightarrow A_4 \wedge A_6$ confidence 2/6

$A_1 \wedge A_2 \wedge A_4 \rightarrow A_6$ confidence 2/2

$A_1 \wedge A_2 \rightarrow A_3$ confidence 3/6

(C)

Figure 2: Association rule discovery

Following above calculations, it is easy to describe a user behavior network, a heterogeneous network of association rules [6], which is used as a user behavior model.

C $(A_1 \wedge A_2 \rightarrow A_7 \wedge A_5)$	50%
C $(A_1 \wedge A_2 \rightarrow A_4 \wedge A_6)$	33%
C $(A_1 \wedge A_2 \rightarrow A_3)$	50%

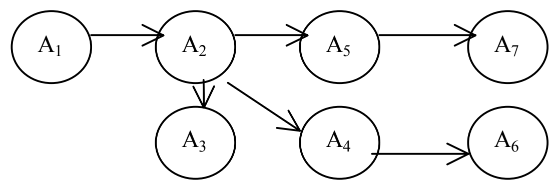

Figure 3. User Behavior Model *confidence: C (x)

2.1.1 *Adjacency and descendant*

Suppose in another transaction set (see Figure 4), between two transactions T5 $\{A_1 \rightarrow A_4 \rightarrow A_7\}$ and T7 $\{A_4 \rightarrow A_1 \rightarrow A_7\}$, some changes of order happens, and the position of A_1 and A_4 is inconsistent, which assumes that there is no strong time sequence between these two actions. Here, a function is used to decide this kind of interrelationship: [the frequency of $(A_{n-1} \rightarrow A_n)$] / [the frequency of $(A_{n-1} \wedge A_n)$] $- 0.5 < t$ (*t* is predetermined as an inconsistent tolerance). If the function is satisfied, the two actions are treated as *adjacency*, and otherwise, they are treated as *descendant*. In this transaction set, the interrelationship between actions A_1 and A_4 is adjacent. The Difference this network from a common association rule network is that the time sequence between actions is considered to describe its topology.

Transaction	Actionset
T1	$A_1 \rightarrow A_2 \rightarrow A_5$
T2	$A_1 \rightarrow A_3 \rightarrow A_4 \rightarrow A_6$
T3	$A_2 \rightarrow A_3$
T4	$A_3 \rightarrow A_4 \rightarrow A_6$
T5	$A_1 \rightarrow A_4 \rightarrow A_7$
T6	$A_2 \rightarrow A_3$
T7	$A_4 \rightarrow A_1 \rightarrow A_7$
T8	$A_1 \rightarrow A_2 \rightarrow A_3 \rightarrow A_5$
T9	$A_1 \rightarrow A_2 \rightarrow A_5$

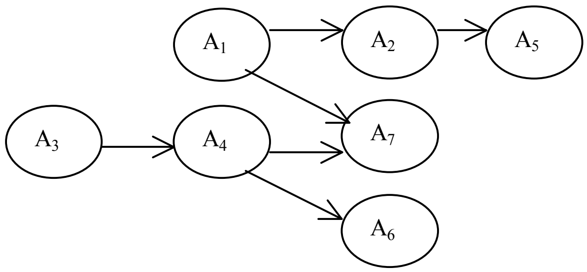

Figure 4. Adjacency and descendant

2.2 The Plan Retrieval Algorithm

Suppose we have an actionset G = $\{A_1 \rightarrow A_2 \rightarrow A_3 \rightarrow A_5 \rightarrow A_7\}$, the confidence $A_1 \rightarrow \{A_2, A_3, A_5, A_7\}$ is $P(G)/P(A_1)$; the confidence $A_1 \wedge A_2 \rightarrow \{A_3, A_5, A_7\}$ is $P(G)/P(A_1, A_2)$, then by following the product rule, $P(A_1, A_2) = P(A_2|A_1)\,P(A_1)$, $P(G)/P(A_1, A_2) = P(G)/P(A_1)/P(A_2|A_1)$.

In a transaction $\{A_1, \ldots, A_n, A_{n+1}, \ldots, A_m\}$ where m>n+1>n>1

$$\textbf{Confidence of } \{A_1, \ldots, A_n\} \rightarrow \{A_{n+1}, \ldots, A_m\} = \frac{\textbf{Confidence of } \{A_1, \ldots, A_{n-1}\} \rightarrow \{A_n, \ldots, A_m\}}{\textbf{Confidence of } \{A_1, \ldots, A_{n-1}\} \rightarrow A_n}$$

If the confidence C_{n-1}: $\{A_1, \ldots, A_{n-1}\} \rightarrow \{A_n, \ldots, A_m\}$ is available, it is easy to get the confidence C_n: $\{A_1, \ldots, A_n\} \rightarrow \{A_{n+1}, \ldots, A_m\}$ by dividing C_{n-1} by the confidence of items $\{A_1, \ldots, A_{n-1}\} \rightarrow A_n$.

According to the above confidence stochastic method, the confidence of an action set is determined by the confidence of its prior action set. Thus, each latest action does not only decide the paths in an association rule network, but also decide the confidence of every path with respect to the current action set.

The process of plan retrieval algorithm is below (the user behavior model see Figure 3):

Step 1: keeping observe an agent's action and record its two continues behaviors, e.g. $\{A_1, A_2\}$.

Step 2: get all transactions containing these two behaviors, e.g.

Actionset	Support count
$\{A_1, A_2, A_5, A_7\}$	2/9
$\{A_1, A_2, A_4, A_6\}$	3/9
$\{A_1, A_2, A_3\}$	3/9

Step 3: check whether the interrelationship between these two behaviors is adjacent or descendent. If descendent, the time sequence in the transaction set must be consistent, e.g. $\{A_1 \rightarrow A_2\}$.

Step 4: get a transaction with the maximum confidence from the list, feedback the following behavior to the agent, e.g. $\{A_1 \rightarrow A_2 \rightarrow A_4 \rightarrow A_6\}$, and feedback the action item A_4.

Step 5: if the agent verifies the feedback, the recognition is successful and the result is the agent's plan, e.g. $\{A_1 \rightarrow A_2 \rightarrow A_4 \rightarrow A_6\}$, but otherwise the assistant keeps observing the agent's coming behaviors and repeat above steps from 3 to 5 for further prediction.

Step 6: after the whole transaction is completed, all of agent's behaviors are recorded into log file to update the user behavior model.

3. SYSTEM APPROACH

In an agent-based virtual world, e.g. e-Market system, each player is represented by a given group of agents, acting as a virtual person. Within this given agent group, an assistant agent undertakes the plan recognition. This assistant keeps observing and recording its corresponding player's manipulations, and further make a prediction.

The agent includes two main function modules: the model maintainer and the plan recognizer. The former, the model maintainer, uses the model building algorithm to build up a user behavior model, and the later, plan recognizer, uses the existing user behavior model to deduct a final action (maybe a goal) and its corresponding plan. Consequently, the agent contains two datasets: a log file and a user behavior model. Being designed to optimize the system performance, the real-time plan recognizer has to react to negotiation agent's behaviors in a short time, based on the existing information. Moreover, to build a user behavior model costs a large of amount of time, but a single transaction affects the model very little. Therefore, once an agent is launched, its model is launched into its working memory, and this model will be updated until the agent's idle time interval.

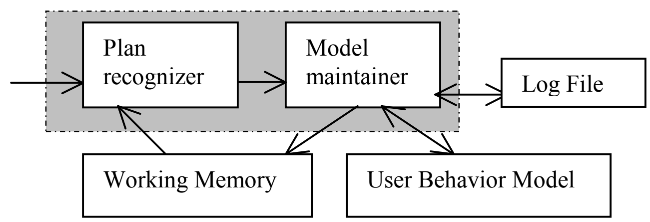

Figure 5. The Main Function Modules

In the section 2, each action is simply represented as an action item, A_n. Usually what sees into the action item depends on the model behind the agent-based virtual world. This model defines the form of behavior of various players. For example, if conceptual graph [8] is used to model user actions in a virtual world, then each action is represented as a sentence, which is constructed by "subject → predicate → object → attribute type → attribute". In a computer game, e.g. "Age of Empire II", a player A attacks a castle of another player B. That action is represented as "User: A→attack→castle→belonging to→User: B". With its advantage, the user's behavior model is able to describe more complex working environment, e.g. virtual e-Market system. Figure 6 shows a single action item presented as a conceptual graph.

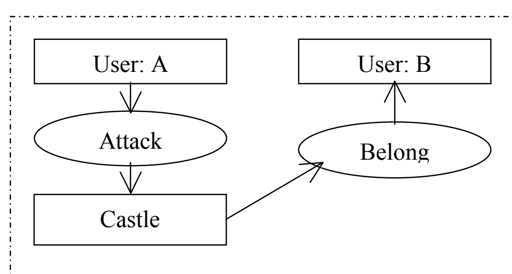

Figure 6. Action Represented as a Conceptual Graph

In such user behavior model, each transaction, the so-called actionset, consists of a series of conceptual graphs, which represents various actions done by players. Current experiments are carried out at the e-market, and the types of players' actions are limited, e.g. price check, buy and sale etc. An assistant captures a player's mouse manipulations and keyboard inputs on an input window as his behaviors, e.g. click, drag and selection etc.

After making successful plan retrieval, the assistant delivers its predicted plan to other members of the agent group. Based on the prediction, they allocate their local resource to the coming player's manipulations, or even respond to some of predicted requests in advance and push the relevant information back to players. Rather than simply waiting for players' actions, these predications enable virtual worlds to behavior "proactivity", i.e. launch some corresponding strategies to optimize their local resource allocation, and response players' requests more efficiently. Among the series of processes, the accuracy and time of the predictions are the key issues undoubtedly.

4. RELATED WORK

Some research on plan recognition has been carried out since the 80's last century, but until recent years, some researchers incorporate uncertainty reasoning into plan recognition. Paek and Horvitz [5] use Belief Networks, Nate Blaylock and James Allen [4] use N-gram Model to represent the likelihood of possible goal and plan. However, Nate Blaylock and James Allen only consider the action types without any flags or arguments. Paek and Horvitz use Belief Network to enable their approach to adapt to a new domain. They consider the detail of the underlying intention of each behavior and human-computer mutual understanding. As a result, their method becomes too complex to be able to respond in real-time.

Neal Lesh, Charles Rich and Candace L. Sidner [3], use a goal and subgoal tree to decompose the possible plan. However, Neal Lesh has to build a goal recipe library manually, which requires a great amount of experts' involvement. Moreover, the recipe library has to be customized for various individual, which greatly affects its practical utility. Practical plan recognition must utilize machine-learning algorithms of some degree to adapt to various users automatically.

5. SUMMARY AND FUTURE WORK

The above presents an initial work on using modified association rules for plan recognition. By setting minimum support values, non-trivial, essential transactions are distilled from a noisy historical dataset. This approach supports the supervised machine learning, which can be plugged into new domains easier than many recognizers.

In the current solution, only the association between user's behaviors frequency is utilized for prediction, without considering their underlying cause and effect relation, their psychological relationship, the impact from outside circumstance, e.g. the layout of a virtual world, and the user's state. Moreover, the action granularity and the isolation between various transactions are other criteria. Too large granularity will lose much information, but too small a granularity will take many noises into system. A method to differentiate several transactions with diverse goals from an action sequence will affect the recognition accuracy greatly.

With the development of environment, similar to the e-market system, more data from different agents is being collected. The analysis of that data will provide more completed user behavior model, and this will assist this approach in the paper to achieve predictions with highly accuracy.

6. ACKNOWLEDGMENTS

My thanks to the members of the e-market group at UTS for assisting with the development of this work

7. REFERENCES

[1] Jung-Jin Lee and Robert McCartney (1998). *Partial Plan Recognition with Incomplete Information*. International Conference on Multi Agent Systems 98(ICMAS'98). URL: http://www.engr.uconn.edu/~jjl/Papers/ICMAS98lee.ps

[2] Jiawei Han and Micheline Kamber (2001). *Data Mining Concepts and Techniques. Chapter 6: Mining Association Rules in Large Databases.* Morgan Kaufmann Publishers.

[3] Neal Lesh, Charles Rich and Candace L. Sidner (1999). *Using Plan Recognition in Human-Computer Collaboration.* International Conference on User Modeling, pps 23-32, June 1999.

[4] Nate Blaylock and James Allen (2003), *Corpus-based, Statistical Goal Recognition*. Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, pages 1303-1308, Mexico, August 9-15 2003. URL: http://www.cs.rochester.edu/u/blaylock/Pubs/Files/ijcai03.pdf

[5] Time Paek and Eric Horvitz (2000). *Conversation as action under uncertainty*. The 16th Conference on Uncertainty in Artificial Intelligence (UAI-2000), Stanford, CA, June 2000

[6] Sanjay Chawla, Bavani Arunasalam, and Joseph Davis (2003). *Mining Open Source Software (OSS) Data using Association Rules Network*. 7th Pacific-Asia Conference, PAKDD03 Seoul, Korea, April/May 2003. URL: http://www.it.usyd.edu.au/~chawla/publications/chawlapakdd03.pdf

[7] Bing Liu, Wynne Hsu, Yiming Ma (1998). *Integrating Classification and Association Rule Mining*. Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98), 80-86, California: AAAI. URL: http://www.comp.nus.edu.sg/~maym/publication/kdd98.doc

[8] James F. Baldwin, Trevor P. Martin and Aimilia Tzanavari (2000). *User Modelling Using Conceptual Graphs for Intelligent Agents*. Conceptual Structures: Logical, Linguistic and Computational Issues (LNCS 1867, Springer) pp 193-206. URL: http://www.enm.bris.ac.uk/research/aigroup/martin/papers00/ICCS.pdf

[9] Margaret H. Dunham (2003). *Data Mining, Introductory and Advanced Topics. Chapter 6 Association Rules*. Pps 166. Prentice Hall.