ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SOFT BEHAVIOUR MODELLING OF USER COMMUNITIES

[1,2] ALFREDO MILANI, [3] VALENTINA FRANZONI

[1] Department of Mathematics and Computer Science, University of Perugia, Perugia, Italy

[2] Department of Computer Science, Hong Kong Baptist University, Hong Kong

[3] Department of Computer, Control and Management Engineering,
Sapienza University of Rome, Rome, Italy

E-mail: [1]alfredo.milani@unipg.it, [2]valentina.franzoni@dmi.unipg.it

ABSTRACT

A soft modelling approach for describing behaviour in on-line user communities is introduced in this work. Behaviour models of individual users in dynamic virtual environments have been described in the literature in terms of timed transition automata; they have various drawbacks. Soft multi/agent behaviour automata are defined and proposed to describe multiple user behaviours and to recognise larger classes of user group histories, such as group histories which contain unexpected behaviours. The notion of deviation from the user community model allows defining a soft parsing process which assesses and evaluates the dynamic behaviour of a group of users interacting in virtual environments, such as e-learning and e-business platforms. The soft automaton model can describe virtually infinite sequences of actions due to multiple users and subject to temporal constraints. Soft measures assess a form of distance of observed behaviours by evaluating the amount of temporal deviation, additional or omitted actions contained in an observed history as well as actions performed by unexpected users. The proposed model allows the soft recognition of user group histories also when the observed actions only partially meet the given behaviour model constraints. This approach is more realistic for real-time user community support systems, concerning standard boolean model recognition, when more than one user model is potentially available, and the extent of deviation from community behaviour models can be used as a guide to generate the system support by anticipation, projection and other known techniques. Experiments based on logs from an e-learning platform and plan compilation of the soft multi-agent behaviour automaton show the expressiveness of the proposed model.

Keywords: *Community Behaviour, User Behaviour, User Interaction, Timed Transition Automaton, Elearning, Automated Planning*

1. INTRODUCTION AND RELATED WORKS

In the research community of user behaviour modelling, several approaches focus on formalising user session and user behaviour based on developing measures, analysing user-action histories or navigation histories [1][2][3][4].

When describing the behaviour of a user group, relationships can be modelled in an explicit way by adding relationship models to individual user models (for example in social networks [5]) or in an implicit way by analysing interactions among users (for an overview see [6]). Most of the existing approaches focus on an a-posteriori analysis of user histories or group interactions to obtain a description of usage patterns of a website or interaction patterns in a virtual environment, with the aim of better addressing target user types or users interacting in a certain way.

In adaptive web systems, a stronger claim is asserted (see [7], in particular [3] and [8]): a real-time community monitoring and user modelling/profiling are needed in order to put into force immediate adaptations. In [2], [9], [10] and [11] some automaton based models are presented in which the user history defined as a sequence of actions are parsed by finite state automata that represent patterns of actions and hence are conceived as user behaviour models. A drawback of such approaches is that they sharply separate the class of user histories conform to the constraints defined by the behaviour model and those deviating from it. In this case, in fact, the behaviour model is merely a “minimum necessary behaviour pattern”,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

and no deviations from it are admitted. Therefore, the information they are able to give about the user is if his/her history is accepted or rejected by the behaviour model automaton and not to give information about how his/her behaviour is near to or far from the behaviour model.

The aim of this work, introduced and explained the following paragraphs, is to define a notion of soft multiagent behaviour automaton which extends the soft user modelling framework described in [10], [11]. We will show how the new framework can be used for community behaviour modelling in a general dynamic virtual system using quantitative examples and experimental results for community behaviour recognition are discussed. Details about the model translation and implementation of planning problems are described in [11].

2. TIMED AUTOMATA FOR USER BEHAVIOUR

2.1 Multiagent Behaviour Automata (MBA)

The use of timed transition automata (TTA) [10] for describing the behaviour of a single user has been proposed in [4], [10], [12].They model the behaviour of the single user operating in a structured interface environment (e.g., e-learning platforms, webmail clients, content management platforms) by a TTA, a state transition diagram extended with time constraints, where each action performed by a user is matched by a state transition label. The timed language accepted by the TTA represents the legal behaviour in the model. In other words, if the sequence of user actions, i.e., the user history, is parsed by the automaton, then the corresponding user behaviour is recognised. A number of drawbacks have been observed on the TTA approach: the model only accounts for the behaviour of a single user not allowing to model for interaction; the parsing process produces a crisp yes/no answer, i.e., exceptions or minor deviations from the ideal behaviour model represented by the TTA are not allowed.

In order to overcome these limits, we introduce the formal notion of soft multiagent behaviour automaton (SMBA), where an SMBA can be viewed as a double extension of TTA: 1) actions are executed by a group of users/agents; 2) the semantics of the parsing process is defined on the notion of deviation measure, i.e. a form of distance of the observed behaviour from the language of histories.

In the following, we first introduce multiagent behaviour automata based on TTA [10] [13], then the soft automata are defined through the concept of deviation measures; the soft parsing semantics and algorithms are finally given. The terms user and agent will be used interchangeably.

Definition 2.1 *Multiagent Behaviour Automaton*. A Multiagent Behaviour Automaton (MBA) is a tuple $(\Sigma, W, S, s_0, C, E)$ where:

- Σ is a set of actions, $W = U \cup V$ with U a finite set of agents/users and V a set of user variables,
- S is a finite set of states, $s_0 \in S$ is an initial state,
- C is a finite set of clocks,
- $E \subseteq S \times S \times \Sigma \times W \times 2^C \times 2^V \times \Phi(C)$ defines the transition table for the automaton.

Each transition $e \in E$ is a 6-ple $e = (s, s^l, a, w, R_C, R_V, \delta)$ representing a transition from state s to state s^l which occurs when action a is executed by agent denoted by w at time τ and clock constraints δ are verified by the current values of clocks; the transition also resets to 0 the subset $R_C \subseteq C$ of clocks, and reset the bindings of variables in RV.

Clocks allow expressing time constraints such as durations relative to subpatterns in the transition diagram. Clocks are initialised to 0, and they are updated as time advances. Clock constraints $\Phi(C)$ are defined by $\delta := x \le c | c \le x | \neg\delta | \delta \wedge \delta$ where $x \in C$ is a clock and c is a rational constant. User variables, denoted by $?u \in V$, allow to refer to different actions executed by the same user without committing to a specific one. During the parsing process, a user variable ?u can be bound to any user constant $u \in U$, the binding holds until a transition reset the variable. By convention user variables are constrained to be bound to distinct users, while symbol ”*” matches any user.

Definition 2.2 *Group History*. A *group history* on a set of actions Σ and a set of agents U, is a finite sequence of triple $[(a0, u0, \tau 0) \ldots (ak, uk, \tau k)]$ where $a_i \in \Sigma, u_i \in U$, $\tau_i \in$R for $i \in [0, k]$ with $\tau_i \le \tau_{i+1}$, $i \in [0, k-1]$

Definition 2.3 *Run of Multiagent Behaviour Automaton*. A *run* of an MBA is a *group history* which defines a sequence of legal state transitions from the initial state s_0.

Note that in this automaton model there is no notion of final acceptance state(or equivalently all states can be considered final) since we are interested in recognising behaviour also from partial observations histories.

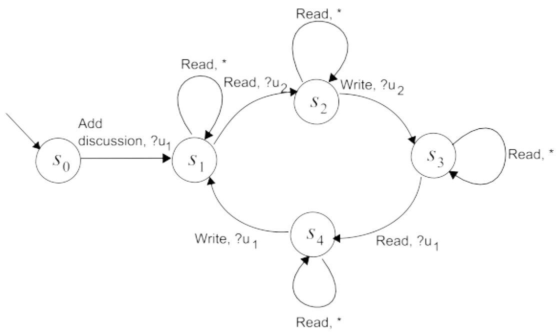


Figure 1:Ideal dialogue automata model in forum

Definition 2.4 *Group Behaviour Model.* The *group behaviour model* defined by an MBA $A = (\Sigma, W, S, s_0, C, E)$ is the set of all group histories which correspond to consistent runs of the automaton starting in s_0, i.e. an history $w = [(ai, uj, \tau i)]$ with $i \in [0, k]$, $uj \in U$ is also $w \in L(A)$ if exists a run from s_0 with each transition $(s, s^l, uj, ai, RC, \delta)$ taking place at time instant τi.

According to automata theory, we can also see a group behaviour model as the timed language L(A) accepted by an MBA.

In Fig.1, Fig.2 and Fig.3, three models of community behaviour in forums are described in terms of multiagent behaviour automata. It is worth noticing that by the $*$ symbol denotes any user in U without the possibility of referencing him/her later; by $?u_i$ we mean a named variable in the set U, that is, a generic user that can be referenced later. Moreover two user variables $?u_i$ and $?u_j$ are supposed to assume different values if $i \neq j$ and equal values if $i=j$. The command Reset $?u_i$ makes the variable $?u_i$ losing its actual value, which becomes then undefined.

2.2 Soft Multiagent Behaviour Automaton

The former notion of group behaviour entails ideal models, which are too strict to account for real situations where the group behaviour does not entirely respect the automaton scheme. In order to relax this requirement, we introduce a more flexible notion of Soft Multiagent Behaviour Automaton(SMBA) which is able to accept a broader set of histories, and rates user group histories according to a measure of deviation from the ideal behaviour model.

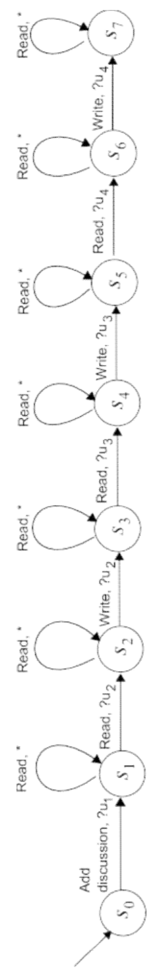


Figure 2. Group forming model for a 4-person group in a forum

For example, the relaxed SMBA would accept group histories with additional *unexpected actions*, with some *missing actions*, with actions which violate the *temporal constraints* or action which are executed by an *unexpected agent*. The degree of deviation from the strict MBA group behaviour model is defined by specific measures which estimate the different kind of violations of the transitions, and assign a different *penalty* when the soft automaton accepts an *illegal* history.

Definition 2.5 *Soft Multiagent Behaviour Automaton.*

A *soft multiagent behaviour Automaton* (*SMBA*) is a tuple

$$A_S = (A, \mu_1, \mu_2, \pi, \sigma, \theta)$$

Journal of Theoretical and Applied Information Technology
15th January 2018. Vol.96. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

where A is an MBA, and μ_1, μ_2, π, σ, θ are deviation measures defined in the following.

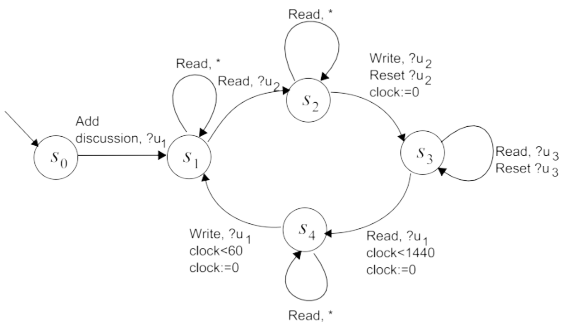


Fig. 3. Moderated Group Behaviour Model in forums

Definition 2.6 *Action Matching*. An *action matching* deviation measure

$$\mu_1: \Sigma \times \Sigma \rightarrow R^+$$

is a function, which assigns a non-negative penalty value $\mu_1(a, b)$ to each pair of action symbols $a, b \in \Sigma$ and where $\mu_1(a, a) = 0$.

Definition 2.7 *User Matching*. A *user matching* deviation measure is a function

$$\mu_2: U \times U \rightarrow R_0^+$$

where $\mu_2(u1, u2)$ assigns a non-negative penalty value to each pair of user symbols

$u_1, u_2 \in U$ and where $\mu_2(u_1, u_1) = 0$.

Definition 2.8 *Time Constraint*. A *time constraint violation* measure is a function

$$\theta: \Delta \times T \rightarrow R_0^+$$

where Δ is the set of the time constraints δ present on the edges in the automaton and T is the set of the time stamps τ in the input history. The measure of *time constraint violation* is based on the fact that a time constraint δ defines a subset of real numbers [14][15][16]. Hence it is possible to check if $\tau \in \delta$ in which case θ=0. Otherwise $\theta > 0$ is defined and calculated as the distance between τ and δ in the real domain.

Since violations of type action matching, user matching and time constraints can occur at the same time, the respective measures are combined into a soft matching measure [17] [18] defined as

$$\mu_1(a, b) + \mu_2(u_1, u_1) + \theta(\delta, \tau)$$

Definition 2.9 *Extra Input Symbol*. An *extra input Symbol* or *skip symbol* deviation measure is a function

$$\pi: \Sigma \times S \rightarrow R_0^+$$

where $\pi(a, si)$ assigns a penalty if in the current input token of the history being parsed there is an extra symbol a without an edge starting from si labelled a. In this case, the parser remains in the same state skipping the current token.

Definition 2.10 *State Skip*. A *State Skip* deviation measure is a function

$$\sigma : T \rightarrow R_0^+$$

where $T = \{(si, sj) \mid si, sj \in S$ and $\exists a \in \Sigma, R_C \subseteq C, \delta \in \Phi(C)$ t.c.$\exists e = (si, sj, a, w, R_C, \delta)\}$.

The function σ assigns a penalty when the action symbol a of the current token (a, u, τ) does not match any legal transition from s_i to s_j, but the parser jumps to s_j without parsing the current token. A transition from s_i to s_j is required to exists in the MBA.

The soft parsing semantic of SMBA based on deviation measures [6], [19]–[23], [16], [24]–[37] can be described by a global deviation γ, initially set to zero and incremented by the appropriate amount of penalties every time a violation of the strict MBA parsing process occurs. Note that regular transitions do not increment γ.

Table 1: Extra Input Symbol.

	Extra Input Symbol Case	
Input word MBA Automaton	↓ _ _ (a,u,τ)_ _ curr_state s_i	↓ _ _(a,u,τ)_ _ curr_state s_i
Notes: Measure of Extra Input Symbol $\pi(a, si)$ is taken and γ is updated: $\gamma := \gamma + \pi(a, si)$ Ex.: $(x, u, \tau)(y, u, \tau)(z, u, \tau) \in L(A)$ $(x, u, \tau)(y, u, \tau)(a, u, \tau)(z, u, \tau) \in LS(A)$ While parsing history (a, u, τ) in input history the current state s_i is left unchanged.		

Definition 2.11 *Behaviour Deviation*. For each history h, a *measure of behaviour deviation* $\gamma(h)$ is defined as the minimal deviation which can be obtained by parsing h in the SMBA.

Journal of Theoretical and Applied Information Technology
15th January 2018. Vol.96. No 1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The extended soft notions of run and behaviour model (or accepted language) of an SMBA are straightforward. Note that differently from MBA, any history is soft- accepted by the SMBA with a specific measure of behaviour deviation [38][39]. It is apparent that the subset of the SMBA accepted language with deviation γ = 0 corresponds to the strict MBA behaviour model.

Table 1, table 2 and table 3 show the main categories of deviation from the model which can occur during parsing.

Table 2: State Skipping.

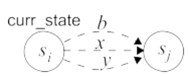
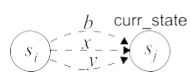
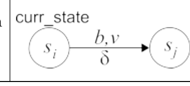
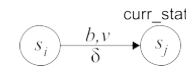
State Skipping Case		
Input word	↓ _ _ (a,u,τ)_ _	↓ _ _ (a,u,τ) _ _
MBA Automaton		
Notes: Measure of State Skip $\sigma(s_i, s_j)$ is taken, and γ is updated: $\gamma := \gamma + \sigma(s_i, s_j)$ Ex.: (x, u, τ)(y, u, τ)(a, u, τ) ∈ L(A) (x, u, τ) (a, u, τ) ∈ LS (A) A transition on the automaton is made from state s_i to state s_j without parsing any symbol within the input history.		

Table 3: Soft Matching.

Soft Matching Case		
Input word MBA Automaton	↓ _ _ (a,u,τ)_ _ 	↓ _ _ (a,u,τ) _ _ 
Notes: Measure of Action and User Matching $\mu_1(a,b)$, and $\mu_2(u,v)$ and Measure of Time Constraint Violation $\theta(\delta,\tau)$ are taken and γ is updated: $\gamma := \gamma + \mu_1(a,b) + \mu_2(u,v) + \theta(\delta,\tau)$ Ex1.: $(x,u,\tau)(b,v,\tau)(y,u,\tau) \in L(A)$ $(x,u,\tau)(a,v,\tau)(y,u,\tau) \in LS(A)$ Symbol a in input history is being made correspond to the symbol b on the edge si-sj . Ex2.: $(x,u,\tau)(b,v,\tau)(y,u,\tau) \in L(A)$ $(x,u,\tau)(b,u,\tau)(y,u,\tau) \in LS(A)$ Agent u in input history is being made correspond to the agent v on the edge si-sj .		

3. THE ALGORITHM

Given a SMBA A_S where s is the current state and $w = (a, u, \tau)$ is the current token of a group history and γ is the current evaluation of the behaviour deviation measure, we build an algorithm *SOFTParse(s,w,γ),* see Fig.4, that returns a set of parsing deviation evaluation values corresponding to the different possible deviations in evaluating a ,

The overall deviation measure [19], [22], [16], [25], [28], [30], [40] for a group history is then:

$$\gamma(w, A_I) = \min\{SoftParse(s_0, w, 0)\}$$

Note that if w is accepted by the corresponding multiagent behaviour automaton A then $\gamma(w, A_S) = 0$, or equivalently, $L(A) \subseteq L(A_S)$.

The pseudocode of the SOFTParse algorithm is shown in Table 4 and Table 5 below.

4 EXPERIMENTS

The SMBA has been implemented using the planning techniques described in [36]–[38] extended to the multi-agent case. The tests have been executed on Intel Pentium IV 3.00GHz with 1GB of RAM running the operating system Linux and using the FF metric planner [42].

Table 4: The SOFTParse Algorithm

```
FUNCTION  SOFTParse(s,w,γ)
REQUIRE
s current state in an SMBA
As,
w  current  token  history,
γ current evaluation of the
behaviour deviation measure
IF  w = tt // tt is terminal
token
RN  γ
//end of parsing
END IF
Let  w := (a0, u0, τ0)w^t
//(a0, u0, τ0) is the current
//token symbol in w
//Non  deterministically
// execute the following
steps
// SEE TABLE 5
//1 - EXACT SYMBOL MATCHING      γ1,j
//2 – SOFT MATCHING              γ2,j
//3 – EXTRA INPUT SYMBOL         γ3,0
//4 – STATE SKIPPING             γ4,j
RETURN  min_i,j{γi,j}
```

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 5: The SOFTParse nondeterministic steps

```
//1 - EXACT SYMBOL MATCHING
//2 – SOFT MATCHING
//3 – EXTRA INPUT SYMBOL
//4  – STATE SKIPPING

//1 - EXACT SYMBOL MATCHING
FOR EACH transitions j
      (s, s^t, a0, u0, Λ, δ) ∈ E
DO
 γ1,j :=
   SoftParse(s^t, w^t, γ + θ(δ,
τ0)

END FOR

//2 – SOFT MATCHING
FOR EACH transitions j
      (s, s^t, a, u, Λ, δ) ∈ E
      with a ≠ a0 OR u ≠ u0
DO
 γ2,j :=
SoftParse(s^t, w^t, γ + μ1(a, a0)+
    + μ2(u, u0) + θ(δ, τ0))
END FOR

//3 – EXTRA INPUT SYMBOL a0
γ3,0 :=
  SoftParse(s, w^t, γ +π(a0, s))


//4  – STATE SKIPPING
FOR EACH transitions j
      (s, s^t, a, u, Λ, δ) ∈ E
 DO
 γ4j :=
 SoftParse(s^t, w^t, γ + σ(s,s^t))
END FOR
```

In the following, some examples are detailed to show how the proposed approach can be applied in an e-learning domain. Let us consider the models of forum community behaviour depicted by means of multiagent behaviour automata in Figures 1, 2 and 3 and the following four histories of user groups [43], shown in table 5, extracted from real e-learning platform's sequences of log files from Seq_1 to Seq_4 where timestamps values are intended in minutes.

The soft parsing of these community histories [44][45][46] by the soft multiagent behaviour automata and their deviation measures are described below. Since all the examples derive from the same domain, the same deviation measures [47][48][49] will be used for the three SMBAs. In particular:

(1) *action matching deviation* measure μ_1 is computed as defined in Table 7;

(2) *user matching deviation* $\mu_2 = 0$ if the agent in the input history corresponds to the user variable defined on the automaton edge, $\mu_2 = 20$ otherwise;

(3) *time constraint violation* measure θ is defined in terms of real distance measured in minutes (time constraints were used only in the group moderator model);

(4) *extra input symbol deviation* measure: $\pi(a, s) = 0$ if a = add discussion and $s = s_0$, $\pi(a, s) = \infty$ if a = add discussion and $s \in S - \{s_0\}$ or the unexpected action is *read* or *write* and $s = s_0$. In remainder cases $\pi(a, s) = 10$ for *read* as extra action and $\pi(a, s) = 30$ for *write* as extra action;

(5) the state skip deviation measure: $\sigma(s_0, s_1) = \infty$, $\sigma(s_i, s_j) = 20$ when the edge s_i - s_j in the automaton is labeled *read* and $\sigma(s_i, s_j) = 50$ when it is labeled *write*..

Table 8: E-learning sample sequences $Seq_{1\text{-}2}$

Seq₁:
[(0, add discussion, A)
(1483, Read,B)
(1484, Write,B)
(2702, Read,C)
(2704, Write, C)
(2705, Read, C)
(2768, Read, A)
(2773, Write, A)
(3864, Read, D)
(3866, Write, D)
(3881, Read, A)
(4083, Read, C)
(9637, Read, C)]

Seq₂:
[(0,add discussion,A)
(1087, Read,D)
(1088, Write,D)
(1720, Read, C)
(1724, Write, C)
(2495, Read, A)
(2496, Read, A)
(2569, Read, T)]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 7 below describes a fragment of the evaluation table used for computing the action matching deviation measure.

Table 7: Action Matching Deviation Measure μ_1 Evaluation Table

$\mu_1(a_i, a_j)$	add discussion	read	write
add discussion	0	∞	∞
read	60	0	10
write	20	30	0

4.1 Discussion of group histories *Seq₁* and *Seq₂*

Analogously, *Seq_2* can be parsed by the Group Forming Behaviour Model represented in figure 3. As in this framework, final states are not previewed, the algorithm ends in s_5 with γ = 0.

4.2 Discussion of group histories *Seq₃* and *Seq₄*

Seq_3 is an example of dialog that can be parsed by the Ideal Dialog Model represented in figure 1. In this case, there is a soft run that will results with γ = 0 ending in state s_4.

See Table 8 for details.

Table 8: E-learning sample sequences Seq_{1-2}

Seq_3:
[(0, add discussion, C)
(348, Read, A)
(349, Read, A)
(352, Write, A)
(444, Read, C)
(1122, Read, A)
(1618, Read, T)
(3161, Read, T)]

Seq_4:
[(0, add discussion, T)
(123, Read, D)
(159, Write, D)
(1605, Read, T)
(1614, Read, T)
(1616, Write, T)
(1678, Read, T)
(2584, Read, C)
(2617, Write, C)
(3944, Read, A)
(4123, Read, B)
(4158, Read, T)
(4290, Read, A)]

Finally, *Seq_4* is an example corresponding to the model of group behavior in a moderated forum (see figure 3) where agent *?u1* is the forum moderator who reads every message within one day and replies to them within an hour. In *Seq_4*, user *T* is the moderator (variable *?u1* takes value *T* and maintains it for the whole parsing process), the forum members conversing with him are first agent *D*, then agent *C* (variable *?u2* takes these two values) [50][51][52]. Other reading members are *A* and *B* (values for user variable *?u3*). The final deviation measure of the best soft parsing of this sequence will be γ = 107 due to time constraint violations (on edge s_3–s_4), and parsing will stop at state s_4.

4. CONCLUSIONS

An original user community behaviour model has been introduced which allows the soft recognition of collective behaviour in user communities also when constraints are partially violated. Soft multiagent behaviour automata are based on extensions of timed automata semantics, where multiple agents can be denoted, and the notions of deviation measure and soft parsing process are introduced in order to assess numerically the number of violations concerning the desired model.

Soft multiagent behaviour automata are defined and proposed to describe multiple user behaviours and to recognise a larger class of user group histories, such as group histories which contain unexpected behaviours.

The notion of *deviation from the user community model* allows defining a soft parsing process which assesses and evaluates the dynamic behaviour of a group of users interacting in virtual environments, such as e-learning and e-business platforms.

The *soft automaton model* can describe virtually infinite sequences of actions due to multiple users and subject to temporal constraints. Soft deviation measures assess a form of distance of observed behaviours by evaluating the amount of temporal deviation, additional or omitted actions contained in an observed history as well as actions performed by unexpected users. The proposed model allows the soft recognition of user group histories also when the observed actions only partially meet the given behaviour model constraints.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The approach proposed in this paper is more realistic for real-time user community support systems, with respect to standard boolean model recognition. When more than one user model is potentially available, and the extent of deviation from community behaviour models can be used as a guide to generate the system support by anticipation, projection and other known techniques.

Experiments based on logs from an e-learning platform and plan compilation of the soft multiagent behaviour automaton show the expressiveness of the proposed model.

Future work will regard the development of an efficient algorithm for soft parsing, the technique of automatic models extraction from observed histories. The future line will investigate techniques used in the area of grammars induction from strings, where the relevant research problem is to generate bottom up a grammar which can parse a given set of input strings. The introduction of time indeed increases the complexity, and a common issue is reducing the overfitting of the automata; it is apparent that is possible to generate such automata by trivially adding a final state for each string in the dataset and a sequence of states corresponding precisely to the sequence of symbols in the strings. The technique of state merging is used in this case to reduce the explosion of the number of states of the automaton, while extensive research will be needed to define what kind of abstraction can be introduced to entail the time constraints embedded in the data.

REFERENCES

[1] M. Teltzrow and B. Berendt, "Web-Usage-Based Success Metrics for Multi-Channel Businesses," in *WebKDD Workshop on Web Mining and Web Usage Analysis*, 2003, pp. 1–11.

[2] S. Ceri, F. Daniel, and F. M. Facca, "Modeling web applications reacting to user behaviors," *Comput. Networks*, vol. 50, no. 10, pp. 1533–1546, 2006.

[3] P. Brusilovsky and E. Millán, "User Models for Adaptive Hypermedia and Adaptive Educational Systems," *Adapt. Web*, pp. 3–53, 2007.

[4] F. Masseglia, P. Poncelet, M. Teisseire, and A. Marascu, "Web usage mining: Extracting unexpected periods from web logs," *Data Min. Knowl. Discov.*, vol. 16, no. 1, pp. 39–65, 2008.

[5] S. Kleanthous and V. Dimitrova, "Modelling semantic relationships and centrality to facilitate community knowledge sharing," in *Lecture Notes in Computer Science*, 2008, vol. 5149 LNCS, pp. 123–132.

[6] S. Akarsh, A. Kishor, R. Niyogi, A. Milani, and P. Mengoni, "Social cooperation in autonomous agents to avoid the tragedy of the commons," *Int. J. Agric. Environ. Inf. Syst.*, vol. 8, no. 2, pp. 1–19, 2017.

[7] P. Brusilovsky, A. Kobsa, and W. Nejdl, *The Adaptive Web: Methods and Strategies of Web Personalization*, vol. 4321. 2007.

[8] A. Soller, "Adaptive Support for Distributed Collaboration," *Adapt. Web*, vol. 4321, pp. 573–595, 2007.

[9] S. Ceri, F. Daniel, V. Demald, and F. M. Facca, "An Approach to User-Behavior-Aware Web Applications," *Int. Conf. Web Eng.*, pp. 417–428, 2005.

[10] A. Milani, J. Jassó, and S. Suriani, "Soft user behavior modeling in virtual environments," in *Proceedings - 3rd International Conference on Convergence and Hybrid Information Technology, ICCIT 2008*, 2008, vol. 2, pp. 1182–1187.

[11] M. Baioletti, A. Milani, V. Poggioni, and F. Rossi, "Experimental evaluation of pheromone models in ACOPlan," *Ann. Math. Artif. Intell.*, vol. 62, no. 3–4, pp. 187–217, 2011.

[12] Q. Wang, H. L. Woo, and J. Zhao, "Investigating critical thinking and knowledge construction in an interactive learning environment," *Interact. Learn. Environ.*, vol. 17, no. 1, pp. 95–104, 2009.

[13] R. Alur and D. L. Dill, "A theory of timed automata," *Theor. Comput. Sci.*, vol. 126, no. 2, pp. 183–235, 1994.

[14] G. Biondi, Y. Li, and "Web-based similarity for emotion recognition in web objects," in *Proceedings of the 9th International Conference on Utility and Cloud Computing - UCC '16*, 2016, pp. 327–332.

[15] S. W. Chan, C. H. C. Leung, and A. Milani, "Knowledge extraction and mining in biomedical research using rule network model," *Lect. Notes Comput. Sci.*, vol. 8211 LNAI, pp. 506–515, 2013.

[16] A. Milani and R. Niyogi, "Automated web services composition with iterated services," *Lect. Notes Comput. Sci.*, vol. 10352 LNAI, pp. 185–194, 2017.

[17] E. Falcinelli, F. Falcinelli, C. Laici, and A.

Milani, “Experience of blended e-learning in post-graduate training for High School Teaching Qualification,” in *Proceedings - International Workshop on Database and Expert Systems Applications, DEXA*, 2007, pp. 663–667.

[18] F. Rossi, and S. Pallottelli, “Planning based integration of web services,” in *Proceedings - 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2006 Workshops Proceedings)*, 2007, pp. 125–128.

[19] Y. Li, P. Mengoni, “Clustering facebook for biased context extraction,” in *Lecture Notes in Computer Science*, 2017, vol. 10404, pp. 717–729.

[20] V. Franzoni, A. Milani, and J. Vallverdú, “Emotional affordances in human-machine interactive planning and negotiation,” in *Proceedings of the International Conference on Web Intelligence - WI ’17*, 2017, pp. 924–930.

[21] A. Chiancone, and “A Multistrain Bacterial Diffusion Model for Link Prediction,” *Int. J. Pattern Recognit. Artif. Intell.*, vol. 31, no. 11, p. 1759024, 2017.

[22] V. Franzoni. “Structural and semantic proximity in information networks,” in *Lecture Notes in Computer Science*, 2017, vol. 10404, pp. 651–666.

[23] G. Biondi, “A web-based system for emotion vector extraction,” in *Lecture Notes in Computer Science*, 2017, vol. 10406 LNCS, pp. 653–668.

[24] M. Baioletti, and V. Santucci, “Algebraic Particle Swarm Optimization for the permutations search space,” in *2017 IEEE Congress on Evolutionary Computation, CEC 2017 - Proceedings*, 2017, pp. 1587–1594.

[25] H. Dee *et al.*, “Playfully coding: Embedding computer science outreach in schools,” in *Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE*, 2017, vol. Part F1286, pp. 176–181.

[26] M. Baioletti, A. Milani, and V. Santucci, “A new precedence-based ant colony optimization for permutation problems,” *Lect. Notes Comput. Sci.*, vol. 10593 LNCS, pp. 960–971, 2017.

[27] V. Santucci, M. Baioletti, “Algebraic differential evolution algorithm for the permutation flowshop scheduling problem with total flowtime criterion,” *IEEE Trans. Evol. Comput.*, vol. 20, no. 5, pp. 682–694, 2016.

[28] L. Mancini, V. Poggioni, and A. Chiancone, “Self regulating mechanisms for network immunization,” in *AI Communications*, 2016, vol. 29, no. 2, pp. 301–317.

[29] A. Chiancone, A. Milani, V. Poggioni, S. Pallottelli, A. Madotto, and V. Franzoni, “A multistrain bacterial model for link prediction andrea chiancone,” in *Proceedings - International Conference on Natural Computation*, 2016, vol. 2016–Janua, pp. 1075–1079.

[30] S. Pallottelli, C. H. C. Leung, and Y. Li, “Context-based image semantic similarity,” in *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2015*, 2016, pp. 1280–1284.

[31] V. Franzoni “A semantic comparison of clustering algorithms for the evaluation of web-based similarity measures,” in *Lecture Notes in Computer Science*, 2016, vol. 9790, pp. 438–452.

[32] V. Franzoni “A pheromone-like model for semantic context extraction from collaborative networks,” in *Proceedings - 2015 IEEE/WIC/ACM International \oint Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT 2015*, 2016, vol. 2016–Jan, pp. 540–547.

[33] M. Baioletti, A. Milani, and V. Santucci, “A discrete differential evolution algorithm for multi-objective permutation flowshop scheduling,” in *CEUR Workshop Proceedings*, 2015, vol. 1493, pp. 80–87.

[34] R. Niyogi “Planning with sets,” *Lect. Notes Comput. Sci.*, vol. 9384, pp. 399–409, 2015.

[35] J. J. Deng, C. H. C. Leung, A. Milani, and L. Chen, “Emotional States Associated with Music: Classification, Prediction of Changes, and Consideration in Recommendation,” *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 1, pp. 1–36, 2015.

[36] V. Franzoni and A. Milani, “Heuristic semantic walk for concept chaining in collaborative networks,” *Int. J. Web Inf. Syst.*, vol. 10, no. 1, pp. 85–103, 2014.

[37] C. H. C. Leung, Y. Li, “Collective evolutionary concept distance based query expansion for effective web document retrieval,” in *Lecture Notes in Computer Science*, 2013, vol. 7974 LNCS, no. PART 4, pp. 657–672.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[38] S. Pallottelli, and A. Milani, "Multi-path traces in semantic graphs for latent knowledge elicitation," in *Proceedings - International Conference on Natural Computation*, 2016, vol. 2016–Janua, pp. 281–288.

[39] E. Gentili, A. Milani, and V. Poggioni, "Data summarization model for user action log files," in *Lecture Notes in Computer Science*, 2012, vol. 7335 LNCS, no. PART 3, pp. 539–549.

[40] S. Akarsh, R. Niyogi, "Modeling socially synergistic behavior in autonomous agents," *Lect. Notes Comput. Sci.*, vol. 9156, pp. 257–272, 2015.

[41] A. Milani and V. Poggioni, "Planning in reactive environments," *Comput. Intell.*, vol. 23, no. 4, pp. 439–463, 2007.

[42] J. Hoffmann and B. Nebel, "The FF planning system: Fast plan generation through heuristic search," *J. Artif. Intell. Res.*, vol. 14, pp. 263–312, 2001.

[43] D. Bartoli, S. Marcugini, and F. Pambianco, "Completeness of the 95256-cap in PG(12, 4)," *Electron. Notes Discret. Math.*, vol. 57, pp. 27–32, 2017.

[44] V. Santucci, "Community of scientist optimization: An autonomy oriented approach to distributed optimization," in *AI Communications*, 2012, vol. 25, no. 2, pp. 157–172.

[45] N. Ukey, R. Niyogi, K. Singh, and V. Poggioni, "A Bidirectional Heuristic Search for web service composition with costs," *Int. J. Web Grid Serv.*, vol. 6, no. 2, pp. 160–175, 2010.

[46] A. Chiancone, Y. Li, K. Markov, "Leveraging zero tail in neighbourhood for link prediction," in *Proceedings - 2015 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT 2015*, 2016, pp. 135–139.

[47] C. H. C. Leung, Y. Li, P. Mengoni, "Set similarity measures for images based on collective knowledge," in *Lecture Notes in Computer Science*, 2015, vol. 9155, pp. 408–417.

[48] V. C. Cheng, C. H. C. Leung, J. Liu, "Probabilistic aspect mining model for drug reviews," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 2002–2013, 2014.

[49] C. H. C. Leung, A. W. S. Chan, i, J. Liu, and Y. Li, "Intelligent Social Media Indexing and Sharing Using an Adaptive Indexing Search Engine," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, pp. 1–27, 2012.

[50] C. Leung, and A. Chan, "Adaptive Search Engines as Discovery Games : an Evolutionary Approach," *Mind*, pp. 444–449, 2008.

[51] G. Biondi, "SEMO: A semantic model for emotion recognition in web objects," in *Proceedings - 2017 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2017*, 2017, pp. 953–958.

[52] E. Falcinelli, C. Gori, J. Jasso, and S. Pallottelli, "E-studium: An Italian experience of blended e-learning for university education support," in *Proceedings - International Workshop on Database and Expert Systems Applications, DEXA*, 2007, pp. 658–662.