Cloning mechanisms to improve agent performances

D. Rosaci and G.M.L. Sarné

DIMET, Università "Mediterranea" di Reggio Calabria, Via Graziella Loc. Feo di Vito, 89060 Reggio Calabria (Italy)
E-mail: {domenico.rosaci,sarne}@unirc.it

Abstract

Learning agents can autonomously improve both knowledge and performances by using learning strategies. Recently, an approach based on a cloning process, called ***EV**olutionary **A**gents* (EVA), has been proposed to obtain more effective recommendations, generating advantages for the whole agent community through individual improvements. In particular, users can substitute unsatisfactory agents with others provided with a good reputation and associated with users having similar interests. This approach is able to support an evolutionary behavior in the community that allows the best agents to emerge over the less productive agents. However, such an approach is user-centric requiring a user's request to clone an agent. Consequently, the approach slowly generates modifications in the agent population. To speed up this evolutionary process, a proactive mechanism called EVA2 is proposed in this paper, where the system autonomously identifies for each user those agents that in the community have a good reputation and share the same interests. The user can check the clones of such suggested agents in order to evaluate their performances and to adopt them. The results of some experiments show significant advantages introduced by the proposed approach.

NOTICE: this is the post-print version of a work that was accepted for publication in Journal of Network and Computer Applications. The editorial version is published in Journal of Network and Computer Applications 36 (1) , pp. 402-408 DOI: 10.1016/j.jnca.2012.04.018

1 Introduction

A learning information agent autonomously and proactively analyzes distributed and heterogeneous information sources for building and updating its knowledge and providing its user with useful recommendations [4, 33, 35]. In other words, a learning agent should be capable to improve its performances in time. Recently, some authors proposed to build communities of intelligent information agents able to modify both their behaviors and their internal knowledge through the use of learning methodologies [26, 31, 32]. For example, in [6] learning agents improve their individual performances by means of a reciprocal mutual monitoring in order to obtain suggestions about the best agents which cooperate and integrate their knowledge. In [32], a user can enrich the knowledge of its agent with that of other agents having similar interests in the community. Differently, other proposals in multi-agent systems (MASs) [8, 39] adopt reputation models rather than similarity measures both to promote agent cooperation and to select the most promising agents for collaboration.

However, while the learning capabilities of an agent produce an improvement in the agent performances, they do not contribute to advantage also the other agents belonging to the same community. On the contrary, biologic "evolution" implies that profitable changes in a population are permanently inherited and spread over the future generations transcending the lifetime of single individuals [15, 17]. In such a way, evolution happens when the genetic material changes from one generation to the next. Differently, occasional changes in individual entities do not produce evolutionary processes.

By considering the peculiarities of both the learning agent systems and the "biologic" environments, in [34] an *evolutionary* framework, called *EVolutionary Agents* (EVA), based on cloning processes and exploiting a reputation model has been proposed. In EVA individual agent's improvements in generating recommendations can induce improvements in the whole learning agent population.

The evolutionary technique adopted in EVA is similar to the biologic asexual reproductive processes generating clones that initially are the exact copies of their parents. On the contrary, in the sexual reproductive processes the parents' DNA are joined to obtain an individual that mixes their characteristics. The nature is mainly oriented on the sexual reproduction because individual changes, in response to environmental changes, are spread on the next generations more quickly than via asexual reproduction. Cloning can be more effective than the sexual reproduction in hostile environments, in presence of strongly selective processes. In this way, the cloning with a suitable mechanism of selection can implement a simple, but effective, mechanism able to induce evolutionary phenomena in a population. In EVA [34] cloning and selection (based on reputation criteria) techniques are adopted in a MAS for allowing a user to require the substitution of unsatisfactory agents with other agents having both similar interests and good reputation in the community.

However, the EVA approach is basically user-centric since it compulsorily requires a user's request for cloning and substituting his/her agent. The consequence is that the evolutionary processes in the agent population occur slowly and, for speeding up them, in this paper it is proposed a new proactive mechanism called EVA2. More in detail, in EVA2 the system autonomously identifies for each user those agents that in the community have a good reputation and share the same interests. Then the user can evaluate the clones of such promising agents in order to compare their performances with those of his/her current agents and, possibly, adopting them.

1.1 Advantages of EVA2

The EVA2 strategy allows to obtain the effectiveness already reached by the previous EVA approach, improving the efficiency of the results. Some experiments performed in a learning agent-based recommender system, and that we will describe in Section 4, confirms that the two evolutionary strategies EVA and EVA2 drastically improve the results in terms of average satisfaction but, moreover, the experiments highlight that EVA2 introduces a significant speed-up. In particular, the result obtained by EVA after 45 days is reached by EVA2 after only 25 days. This reduction of temporal cost is clearly due to the use of the proactive mechanism of agent selection implemented by EVA2, that helps the user to choose the agents to substitute.

Another experimental analysis shows that EVA2 seems to work well by testing at each session a number of new agents closed to 50 percent of the size of the agent set, while the users that test a too low or too high number of agents obtain less effective performances.

The paper is organized as follows. Sections 2 presents an overview of the EVA framework, while Section 3 introduces the new evolutionary strategy EVA2. An evaluation of EVA2 is presented in Section 4, while some related work about mutual agent monitoring is provided in Section 5. Finally, in Section 6 some conclusions are drawn.

2 Overview of EVA

This section presents an overview of the EVA framework. The basic idea exploited in EVA is that, in presence of an unsatisfactory agent, a user can require the system to provide him/her with one or more suitable and performing agents. For each agent in the EVA framework, the system computes a score based on both the similarity with the user's interests and its reputation (considered likely to a genetic component) in the community. The agents having the best scores are cloned and sent to the requester user. In the following, let u be a generic user belonging to the users' community U and assisted by a set $A_u = \{a_i \mid i = 1 \cdots n_u\}$ of n_u information software agents a_i supporting his/her Web activities with recommendations.

2.1 Computing the user's satisfaction

For each Web page visited by u, each agent a_i generates for him/her some suggestions (i.e., Web links). Considering the life of a_i, let R_i and L_u be the sets, partially overlapping, of the Web links suggested by a_i to u and those selected by u, respectively. To evaluate the quality of these recommendation sets, *precision* and *recall* measures [19, 29] have been used. Precision is the fraction of the recommendations considered as relevant by u with respect to the potentially relevant recommendable links stored in L_u. Recall is the fraction of the links actually selected by u and successfully recommended by a_i but alone it is meaningless because returning all possible links as recommendations it is equal to 1. A good recommender agent should have both high precision and recall values. Precision and recall of R_i can be formally defined as:

$$Pre(R_i) = \frac{|R_i \bigcap L_u|}{|R_i|}; \qquad Rec(R_i) = \frac{|R_i \bigcap L_u|}{|L_u|} \tag{1}$$

To consider together recall and precision, their harmonic mean, known as *F-measure* [44] is used. Weighting the precision with respect to the recall, it is obtained the more general F_β-measure, where β is a non-negative real:

$$F_\beta(R_i) = (1 + \beta^2) * \frac{Pre(R_i) * Rec(R_i)}{\beta^2 * Pre(R_i) + Rec(R_i)} \tag{2}$$

In EVA precision, recall and F_β measures are adopted to compute the satisfaction of u for the recommendations provided both by his/her agent a_i in R_i and by his/her whole agent-set A_u by considering the union of the sets R_i relative to each agent $a_i \in A_u$. Formally:

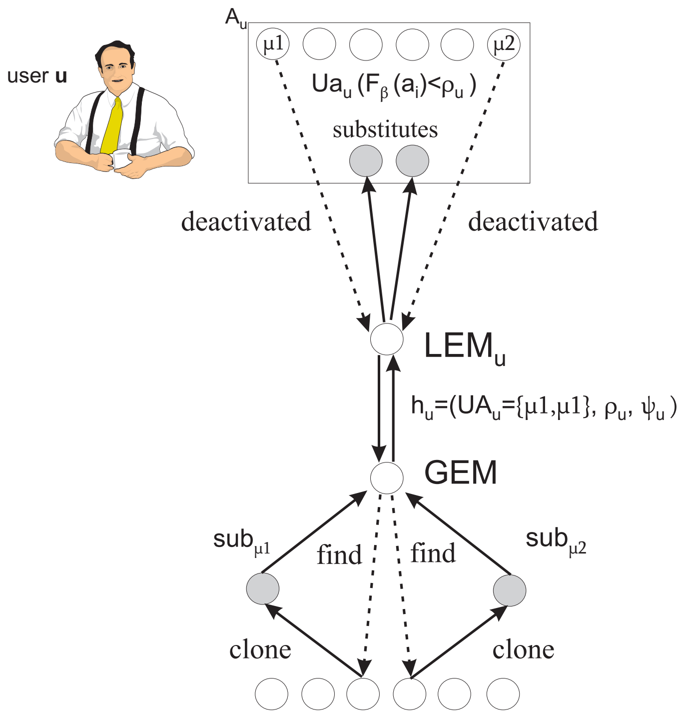

Fig. 1. The evolutionary strategy of the EVA framework

$$Pre(A_u) = \frac{|\bigcup_{i=1}^{n(u)} R_i \cap L_u|}{|\bigcup_{i=1}^{n(u)} R_i|}; \quad Rec(A_u) = \frac{|\bigcup_{i=1}^{n(u)} R_i \cap L_u|}{|L_u|} \tag{3}$$

$$F_\beta(A_u) = (1 + \beta^2) * \frac{Pre(A_u) * Rec(A_u)}{\beta^2 * Pre(A_u) + Rec(A_u)} \tag{4}$$

Furthermore, the F_β-measure is adopted to synthetically evaluate the user's satisfaction simply by observing the acceptance of the provided recommendations.

2.2 Cloning mechanisms to improve user's satisfaction

The EVA framework (depicted in Figure 1), to increase the users' satisfaction about the agents, implements an evolutionary strategy managed by two types of agent, namely: *i)* the *Local Evolution Manager* (LEM_u) agent associated with each user u; *ii)* the *Global Evolution Manager* (GEM) agent associated with the Multi Agent System. The evolutionary strategy is based on the following ideas:

- The satisfaction of a user u for the suggestions provided by his/her agent-set A_u is measured by $F_\beta(A_u)$.
- Each user u can arbitrarily set both the coefficient β, used in computing

$F_\beta(A_u)$, and the *satisfaction threshold* ρ_u for $F_\beta(A_u)$ under which u is unsatisfied of the recommendations generated by his/her agent-set.

- For each user u his/her LEM_u agent periodically computes $F_\beta(A_u)$. If $F_\beta(A_u) < \rho_u$ then LEM_u: *i)* identifies the set UA_u of the unsatisfactory agents for which $F_\beta(a_i) < \rho_u$; *ii)* deactivates the agents belonging to UA_u; *iii)* sends a triplet $\langle UA_u, \rho_u, \psi_u \rangle$ (with the set UA_u, the threshold satisfaction ρ_u and the parameter $\psi_u \in [0.0, 1.0]$, that represents how much the user u weights the similarity with respect to the reputation) to the GEM agent; *iv)* requires the substitution of the deactivated agent with other, presumably more satisfactory, to the GEM agent. The GEM agent (see below) will determine a set of substitutes agents based on both their *reputation* in the community and the similarity (represented by ψ_u) with the deactivated agents. For example, if $\psi_u = 0.3$ the user gives a 30% of relevance to the similarity and a 70% of relevance to the reputation.
- The GEM agent maintains a similarity matrix $\Sigma = \{\Sigma_{i,j}\}$, $i, j \in MAS$ where each element belongs to $[0.0, 1.0]$ and represents the similarity between two agents of the MAS computed as in [32]. Moreover, for each agent $a \in MAS$ the GEM agent stores a *reputation coefficient* $r_a \in [0.0, 1.0]$ (see Section 2.3) that represents a measure of how much the community considers satisfactory the performances of a. When GEM receives the LEM_u request (i.e., $\langle UA_u, \rho_u, \psi_u \rangle$), it inserts in the set C_μ those agents of the MAS having $F_\beta > \rho_u$ with which to substitute each agent $\mu \in UA_u$. Then, GEM computes for each agent $a \in C_\mu$ the following score:

$$s(a, \mu) = \psi_u \cdot \Sigma_{a,\mu} + (1 - \psi_u) \cdot r_a \tag{5}$$

  and, based on it, chooses as substitute of μ the agent sub_μ with the best score (in the case of equal score, the agent having the best F_β-measure will be chosen).
- The GEM creates, for each agent $\mu \in UA_u$, an agent sub^*_μ cloned by the substitute agent sub_μ and having the same ontology. Similarly that in [32], the ontology of an information agent contains both its categories of interests and the causal implications (i.e., relationships between the considered events) learnt by it during its life. Thus, cloning is the duplication of this information as in the nature is duplicated the genetic material. The clone agent sub^*_μ is then transmitted to the LEM_u agent in substitution of the unsatisfactory agent μ. From now the agent sub^*_μ will be completely independent from its parent μ living in the environment of another user. This way, the agent sub^*_μ monitoring the activity of u probably it will modify its initial personal ontology with new information.

Summarizing, the strategy of EVA consists in permitting to a user u of substituting each his/her unsatisfactory agent μ with another agent $sub^*_\mu \in MAS$

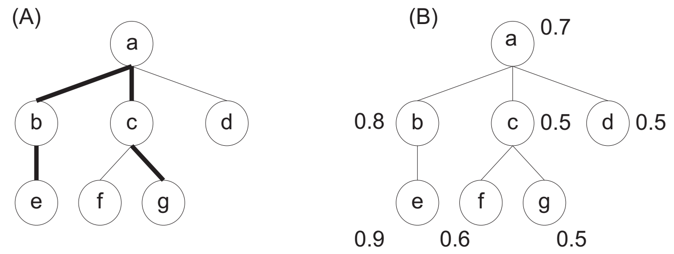

Fig. 2. An example of Descent Tree

based on a cooperation between the agents LEM_u and GEM. This substitution should advantage the user u being sub^*_μ the clone of an agent with: *i)* a F_β-measure (computed by its own user) greater than the u's satisfaction threshold ρ_u; *ii)* a top score, computed based on both its reputation in the MAS and its similarity with the substituted agent. The first property assures that the parent agent of sub^*_μ satisfies its own user but not that its clone will produce an F_β-measure satisfactory for u that has a different perception of the satisfaction. The second one guarantees both that the parent agent of sub^*_μ has a good reputation in the community and that its personal ontology is similar to that of the agent μ. Together, these properties provide u with new agent-sets potentially able to improve the $F_\beta(A_u)$ measure.

2.3 Agent's reputation in the EVA environment

In MASs the reputation (i.e., the opinion of an agent about something [9]) has been studied in a lot of models and surveys (see Section 5) and, accordingly with [38], three main issues are recognized: *i)* reputation of an agent is a multi-dimensional concept (For instance, the reputation of a good eBay seller summarizes those of having good products, applying suitable prices, giving appropriate products descriptions, providing fast and secure delivery, etc.); *ii)* each agent has a different *ontological* dimension of the reputation (i.e., it weights each aspect of the reputation differently based on its personal point of view); *iii)* in a MAS there are an *individual* (for each agent) and a *social* (for the MAS) dimension of the reputation.

In particular, in EVA the individual dimension of the reputation is only that to provide effective recommendations to the agent's owner and the social dimension is the cloning activity (remember that an agent can be cloned and its clones supporting other users). As possible ontological dimensions (see Section 2.1) both the precision and the recall of the recommendations can be identified. Consequently, as a global measure of the individual reputation of the agent a is adopted the $F_{\beta_{u_a}}(a)$ measure that considers both the two ontological dimensions (u_a denotes the owner of a and β_{u_a} the quantitative representation of the consideration of u_a for the precision with respect to the recall).

The agent reputation has also to consider that the evolutionary strategy implies a cloned agent is moved in a new environment. The relationships introduced by the cloning in the set of agents are described by the same terminology adopted to represent genealogical relationships. For instance, in Figure 2-(A) a "genealogical" tree represents a set of agents, associated with the nodes, involved in cloning processes, associated with edges, and where a *parent* is the agent cloned and a *child* is one of its clones. Furthermore, it is possible to define the following formal definition:

Parent and Sibling Agent - Let a be an agent of the community. We denote by $children_a$ the set of one or more clones of this agent. Two agents b and c, both belonging to $children_a$, are called *sibling agents*. Correspondingly, a is called the *parent agent* of each agent belonging to $children_a$

Ancestor Agent - Let a and p be two agents of the community. We say that p is an *ancestor agent* of a if either: *i)* p is the parent agent of a, or *ii)* recursively there is an agent c in the community such that a is a descendant of p via c.

Relatives, Descent Tree and Kinship Degree - Let a and b be two agents of the community. We say that a and b are *relatives* if they share a common ancestor agent p. We call *family* of a, denoted by $\mathcal{F}_a$ the set of all the relatives of a. We define the *Descent Tree* of a, a tree $DT_a = \langle V, E \rangle$ such that *i)* each agent $x \in \mathcal{F}_a$ is associated with a unique vertex $v_a \in V$ and *ii)* each pair (x, y), $x, y \in \mathcal{F}_a$, such that x is the parent agent of y, is associated with a unique edge $e_{x,y} \in E$. Finally, let a and b be two agents, such that they are relatives. We define the *kinship degree* of a and b, denoted by $k_{a,b}$, the length of the path that links a and b in the Descent Tree DT_a.

As a consequence:

(1) At the cloning time, each clone b of an agent a (i.e., $b \in children_a$) is identical to a and inherits its reputation.
(2) Since b supports a user, different from that of a, its initial inherited reputation will evolve in time taking into account also the satisfaction degree of its current owner. The inherited reputation and the individual satisfaction are combined in a unique, global, measure of reputation.
(3) For the cloning processes, each agent a belongs to a *family* of relatives (i.e., the descent tree DT_a) with which a shares some similarities inherited from the cloning process and that affect its performances. This introduces a social component in the computation of the reputation.

These observations are summarized in the *reputation coefficient* r_a associated with each agent a, with $r_a \in [0.0, 1.0]$ (where 1.0 means a complete reliability of a). This coefficient is weighted using the F_β measures of all the n agents belonging to the descent tree DT_a. Each contribute due to an agent b

is weighted in a decreasing manner, based on the kinship degree k between a and b in DT_a, by a coefficient equal to $1/(k_{a,b}+1)$. This way, the contribution to the satisfaction obtained by each other relative is as smaller as higher is the kinship degree with respect to a. More formally:

$$r_a = \frac{\sum_{b \in \mathcal{F}_a} \frac{F_{\beta_b}(b)}{k_{a,b}}}{\sum_{b \in \mathcal{F}_a} \frac{1}{k_{a,b}}} \tag{6}$$

For example, in Figure 2-(B), the agent e has a F_β-measure (i.e., satisfaction) equal to 0.9 but a reputation of 0.696.

3 The novel EVA2 strategy

To speed up the evolutionary process in the agent community a new strategy has been implemented. More in detail, in this new approach, the GEM agent *i)* has to satisfy the user's request to substitute his/her unsatisfactory agents, as in the native EVA strategy,and *ii)* proposes to the user of testing those agents that potentially could enter in his/her agent set in substitution of other agents or in addition to them. In order to perform this proactive mechanism, the native EVA strategy presented in Section 2.2 is modified as follows:

- The information that the LEM_u agent of each user u sends to the GEM agent are now represented by a tuple $\langle UA_u, \rho_u, \psi_u, T_u, N_u \rangle$ where the first three parameters have the same meaning described in Section 2.2, while T_u and N_u are two u's parameters that respectively specify the time (expressed in days) between two consecutive test sessions and the number of agents, ranging in $[0; N_g]$, that u desires to test for each test session (Note that 0 means that u does not want to test any agent, while N_g is the maximum number of agents to test in a single test session and it is a system parameter).
- The GEM agent exploits its similarity matrix Σ and the agents' reputation scores to select for each user u, accordingly to his/her parameters ρ_u, ψ_u, T_u and N_u, a set of agents to clone for a new u's test session.
- After each test session the LEM_u agent evaluates the performances of each clone proposed by the GEM agent. For the agents that really increase the user's satisfaction they can be added to the own agent-set A_u or substitute the less performing agents in A_u.

4 Experimental results

In this section some experiments devoted to test in a MAS the novel strategy implemented in EVA2 are presented. Experiments have been carried out, similarly to that performed to evaluate the native EVA strategy (see 2.2) in [34]), on the top of the CILIOS recommender system [32] for suggesting Web pages to users. In particular, each recommended Web page i is associated with two rates, ranging in $[1, \cdots, 5]$, to represent both the relevances of i for the user evaluated by the system (p_i) and explicitly provided by the user after his/her visit to i (r_i).

The experiments have involved two sets $S1$ and $S2$, each composed of 75 real users. The users of $S1$ adopt the new EVA2 strategy, while those of $S2$ use the old EVA strategy. Each user is provided with a set of 10 agents, and the evolutionary strategy EVA2 has been set for testing 5 agents in the agent set of each user (i.e., $N_u = 5$). A set of XML Web sites publicly available at [27] have been exploited and each agent has been provided with a personal ontology, like to that in [32], using the concepts stored in [27]. Each agent is a CILIOS agent, while the MAS is managed by a GEM agent. The average satisfaction $AS(S)$ of the users belonging to each set S is computed as $AS(S) = \frac{1}{|S|} \cdot \sum_{u \in S} F_1(A_u)$ (we have chosen to use the F_β-measure with a value $\beta = 1$).

The values of $AS(S)$ obtained in the tests are shown in Figure 3. In this figure, the noEV line represents the average satisfaction computed on the users belonging to $S1 \bigcup S2$ after the first five days, without activating any evolutionary strategy. The curve EVA2 (resp. EVA) represents the average satisfaction $AS(S1)$ (resp. $AS(S2)$) obtained activating the strategy EVA2 (resp. EVA) for several days.

The results clearly show that the two evolutionary strategies EVA and EVA2 drastically improve the results in terms of average satisfaction. Moreover, we see that EVA2 introduces a significant speed-up in improving the users' satisfaction. In particular, the result obtained by EVA after 45 days (AS=0.83) is reached by EVA2 after only 25 days. This reduction of temporal cost is clearly due to the use of the proactive mechanism of agent selection implemented by EVA2, that avoids the user manually chooses the agents to substitute.

In order to study the effect of the number N_u of agents that a user u can substitute, we have performed a second experiment, in which we have monitored four set of users, denoted by $S3$, $S4$, $S5$ and $S6$, each composed of 25 users, where the users of each set exploits a different value of N_u. More in particular, the users of $S3$ (resp. $S4$, $S5$, $S6$) exploits a value $N_u = 2$ (resp. $N_u = 4$, $N_u = 6$, $N_u = 8$). In Figure 4 we have plotted the average satisfaction AS of each set for different values of time.

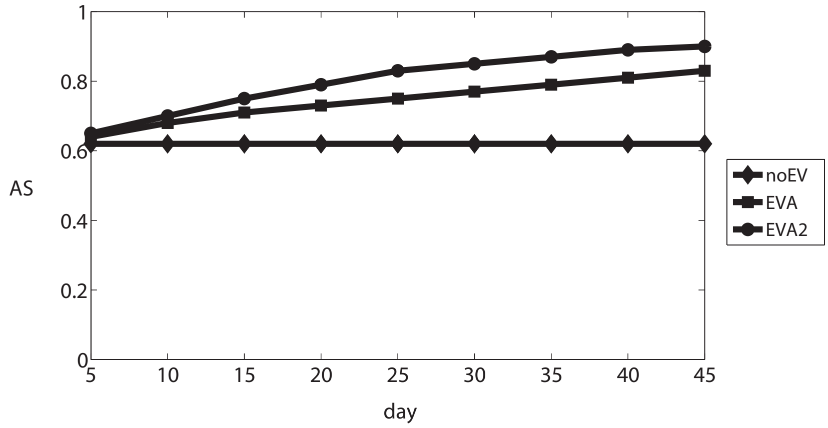

Fig. 3. Comparison between EVA and EVA2 in terms of average satisfaction AS

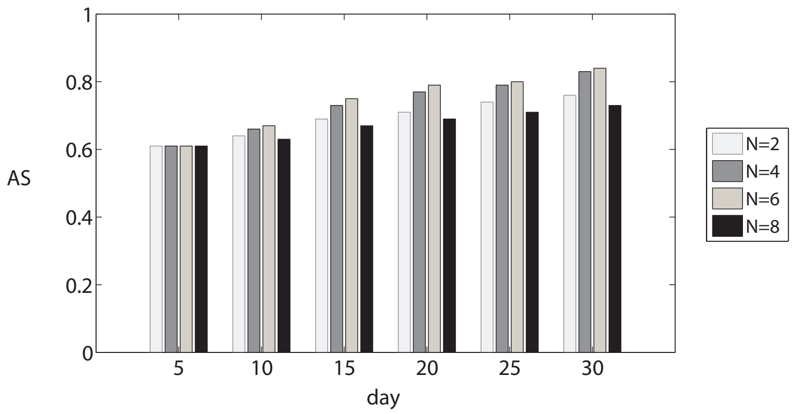

Fig. 4. Average satisfaction using EVA2 for different values of N

The figure shows that the best satisfaction is obtained in the set of the users exploiting $N_u = 6$, while the second best score is obtained by the users exploiting $N_u = 4$. The users that exploit very low or very high values of N_u (i.e., $N_u = 2$ and $N_u = 8$) obtain less effective performances.

We argue that EVA2 seems to work well by testing at each session a number of new agents closed to 50 percent of the size of the agent set.

5 Related Work

Nowadays, multi-agent systems (MASs) are widely recognized as a consolidated approach to deal with dynamic environments [25], particularly in the case of large systems [1], where a number of models and architectures have been proposed [41, 43]. In such a context, in order to achieve its goals a MAS has to adapt both its structure and its behaviors to the changes produced in the environment. This implies continuous modifications in the organization of the MAS and/or in the composition of the agent population in terms of roles, groups, tasks and interactions. In the biologic sense these modifications can be considered an "evolutionary process" of adaptivity in presence of environmental changes [15]. In the remaining of this section we overview some approaches that, to the best of our knowledge, are close to our proposal.

In the aforementioned scenario, a relevant issue is represented by the activity of monitoring learning agents. In this perspective, an evolutionary process means to learn and keep up with a dynamic changing world leaving each agent to interact with each other [42] for maximizing some utility factors. The results of such a process are not predictable but an agent could be considered effective in its society only when its utility grows in time [13].

Thus, each learning agent, in order to suitably support its user in a personalized manner, should be provided with an internal representation about his/her interests and behaviors. To this aim, an ontology representing knowledge of a given domain is usually exploited to capture information between concepts and their relationships [2, 3, 14, 32]. To select the best agents for knowledge-sharing purpose, agents can exploit a common ontology or, in open MASs, several individual ontologies, and in this case inter-ontology properties should be detected to provide agents with a common language.

Some MASs provided with a common ontology are presented in [6, 18]. In the system described in [6], the semantic properties of similarity and complementarity are modeled by parameters. They are taken into account by some adaptive algorithms together with past and current user's choices for suggesting to him/her a set of cooperative agents as closer as possible to his/her expectancies. In [18] authors design collaborative agents for learning and adapting processes to model dynamic users' profiles for realizing profitable interactions. The user's browsing behavior is implicitly tracked by means of a personalized search system to extract his/her short and long term interests. The interests are represented by ontological concepts (common to all the agents) constructed by mapping to a reference ontology the Web pages visited by each user. The approach proposed in [32] induces logical rules to represent agent's behavior in the ontology by means of a connectionist ontology representation, derived from [10], based on neural-symbolic networks. Here the mutual monitoring is realized by introducing a similarity measure of the agent ontology, also considering a logical representation of the agent's behavior. In the systems presented in [7,12,16] several agents' ontologies coexist and discrepancies among individual concepts for synonymies and homonymies are solved for allowing agents' cooperation. Agents cooperate in order to recommend resources by considering similarities between user profiles, which reflect social and semantic features existing in the system.

In the aforementioned works, when the knowledge of an agent is inadequate to support its user, it can be enriched with that of other agents via: *i)* knowledge integration, as in [6]; *ii)* direct interaction, as in [18]; *iii)* learning from logical rules, as in [32]. The definition of inter-ontologies properties is preventively required in presence of different ontologies, as in [7, 12, 16]. These processes permanently modify the knowledge of the involved agent. Differently, in EVA knowledge evolves *i)* in each agent only monitoring the behavior of the current user and *ii)* in the overall agent set by adding or deleting clones from it. Moreover, the learning activity realized in [12] can improve in time the effectiveness of the agent but, differently from our approach and those presented in [6, 18], this improvement does not introduces in the whole system a cooperative behavior among the agents.

Although the similarity among agents plays a main role to determine the most appropriated agent to interact [20], other semantic properties, also reciprocally combined as in [6], can be exploited to obtain a refined knowledge. In large and dynamic agent societies, some authors consider the possibility to adopt a reputation system [45] to support future agent interactions. Reputation models, in order to represent the trustworthiness of an agent, considers a lot of malicious behaviors possibly adopted to avoid a correct perception of the reputation (the interesting reader can refer to [9, 22, 28, 30, 37, 39] for a more comprehensive overview).

In many models the different reputation sources are often evaluated by weighing their credibility [9,21,36,38], usually represented by their own reputation while in EVA is used the F_β-measure. Furthermore, in the "genealogic" reputation approach of EVA the contribution to the reputation of an agent due to its ancestors is weighted by genealogic distances. This way, the agent's ancestor that are closest to the involved agent has a larger relevance in determining the agent's reputation. At the best of our knowledge, no other system adopts this approach to obtain a sort of "forgiveness" effect, commonly based on temporal considerations [24]. However, other decay laws could be exploited as in CellTrust [21] where it is adopted an exponential decay law that is time independent.

The cloning process represents a possible way to solve different problems including, for example, searching, knowledge transfer, overload conditions, failure risks and adaptivity to changes requiring the reorganization of a MAS. Frequently this technique is exploited in a distributed environment, leaving the whole cloning task on the agents' shoulders, while centralized approaches are rarely proposed in the literature.

Cloning of personal agents are presented in [11, 40] for load balancing reasons. Agents rationally detect when *i)* they are overloaded, *ii)* resources are available and *iii)* clones produce beneficial effects in the system. When such constraints are satisfied, an agent autonomously clones itself and gives to its clone a part of its tasks. In [40] an agent could be generated locally on its machine and then migrated on another machine or directly cloned on the remote computer. Birukou and al. [5] presents a MAS for facilitating scientific publications search. To generate effective recommendations, each user's agent, in the interest of other users, describes in behavioral patterns how its owner uses publications. Agents share their knowledge and clone themselves to transfer their knowledge upon request of another less expert agent. A recent rare centralized cloning architecture is developed in [23] to support users in reports writing to reduce their information contribution. Agents are mobile and write a certain amount of text automatically and independently. Agents work locally and on a network without producing identical reports, even in presence of identical contents provided by the user.

The presented cloning examples involve personal agents as EVA. In [11, 40], cloning is generated by autonomous agents' choices, while in [5] it is due to a request of another agent and in [23] it is decided by a central entity. Instead, in the EVA architecture the cloning task raises by a different mechanism and it is carried out by means of the interaction between a personal agent (i.e., the LEM agent) and the central agent (i.e., the GEM agent).

6 Conclusions

EVA is an evolutionary agent system based on a cloning process that allows a user to increase the own satisfaction level. EVA, in its first version, allows an unsatisfied owner of his/her agents can require to the system of providing him/her with clones of those agents belonging to the community that are considered similar for interests to the requester user, having a good reputation in the whole agent community and potentially effective for him/her. As a consequence, individual agent improvements in providing recommendations involve the whole agent community supporting an evolutionary behavior and allowing the better agents to predominate in time over the less productive agents. The core of the EVA strategy is a reputation model, where a clone agent initially inherits the reputation of its parent agent and then it will autonomously evolves in its own environment, using its learning capabilities to increase this "genetic", initial contribution to its reputation. However, this approach slowly produces changes in the agent population. This characteristic is intrinsic of the exploited user-centric approach that needs of a user's request to clone an agent.

To provide a solution to the problem of speeding up the evolutionary process implemented in EVA, in this paper a novel proactive strategy, called EVA2, is presented. In particular, the system, autonomously and accordingly to the user's preferences, selects those agents candidates to potentially improve the performances of the agent-set supporting the user. Periodically, some agent clones are proposed by the system to the user, for a test session. After each test session those agents that really increase the performances could be added to the user's agent set, or substitute the less performing agents. To verify if this new strategy effectively promotes evolution more efficiently than the native approach, an experimental campaign has been realized and the results have been evaluated. The experiments have confirmed the effectiveness of the novel approach showing that the performances increase more quickly with respect to the previous EVA approach.

References

[1] AA.VV., *Software Engineering for Self-Adaptive Systems: A Research Roadmap*, ser. LNCS, Ed. B. Cheng, R. de Lemos, H. Giese, P. Inverardi and J. Magee, vol. 5525, pp. 1–26, 2009.

[2] A. Abecker and L. Elst, *Ontologies for Knowledge Management. Handbook on Ontologies*, Springer, pp. 713–734, 2009.

[3] M. Abramson and R. Mittu, "Learning and coordination: An overview", in *Proc. of the 2011 Int. Conf. on Collaboration Technologies and Systems (CTS 2011)*, pp. 343–350, 2011.

[4] S. Bergamaschi, P. Edwards and P. Petta, *Intelligent Information Agents*, ser. Lecture Notes in Computer Science, vol. 2586, 2003.

[5] A. Birukou, E. Blanzieri and P. Giorgini, "A Multi-Agent System that Facilitates Scientific Publications Search", in *Proc. of the 5th Int. Joint Conf. on Autonomous Agents and Multiagent Systems, (AAMAS '06)*, pp. 265–272, 2006.

[6] F. Buccafurri, L. Palopoli, D. Rosaci and G.M.L. Sarné, "Modeling Cooperation in Multi-Agent Communities", *Cognitive Systems Research*, 6(3), pp. 171–190, 2004.

[7] F. Buccafurri, D. Rosaci, G.M.L. Sarné and D. Ursino, "An Agent-Based Hierarchical Clustering Approach for e-Commerce Environments", in *Proc. of the 3rd Conf. on e-Commerce and Web Technologies, (EC-Web2002)*, ser. LNCS, vol. 2455, pp. 115–118, 2002.

[8] J. Carbo, J.M. Molina and J. Davila, "Comparing Predictions of SPORAS vs. a Fuzzy Reputation Agent System", in *Proc. of 3rd Int. Conf. on Fuzzy Sets and Fuzzy Systems*, pp. 147–153, 2002.

[9] R. Conte and M. Paolucci, *Reputation in Artificial Societies: Social Beliefs for Social Order*. Kluwer Academic Publishers, 2002.

[10] A.S. d'Avila Garcez and G. Zaverucha, "The Connectionist Inductive Learning and Logic Programming System", in *Proc. of the 3rd Int. Conf. on Fuzzy Sets and Fuzzy Systems*, pp. 59–77, 2002.

[11] K. Decker, A. Pannu, K. Sycara and M. Williamson, "Designing behaviors for information agents", in *Proc. of the 1st Int. Conf. on Autonomous Agents (AGENTS '97)*, pp. 402–412, 1997.

[12] P. De Meo, D. Rosaci, G.M.L. Sarné , D. Ursino, and G. Terracina *EC-XAMAS: Supporting e-Commerce Activities by an Xml-Based Adaptive Multi-Agent System*, Applied Artificial Intelligence, 21(6), pp. 529–562, 2007.

[13] V. Dignum, F. Dignum and L. Sonenberg, "Towards Dynamic Reorganization of Agent Societies", in *Proc. of the 3rd Work. on Coordination in Emergent Agent Societies*, pp. 22–27, 2004.

[14] V. Eyharabide and A. Amandi *Ontology-based User Profile Learning*, Applied Intelligence, pp. 1–13. http://dx.doi.org/10.1007/s10489-011-0301-4, 2011.

[15] D.J. Futuyma, *Evolution.* Sunderland: Sinauer Associates, 2005.

[16] K. Juszczyszyn, P. Kazienko and K. Musial, "Personalized Ontology-Based Recommender Systems for Multimedia Objects", in *Agent and Multi-agent Technology for Internet and Enterprise Systems*, Eds. A. Hkansson, R. Hartung, and N. Nguyen, ser. LNCS, vol. 289, pp.275–292, 2010.

[17] S.J. Gould, *The Structure of Evolutionary Theory.* Belknap Press, 2002.

[18] A. Hawalah, M. Fasli, "A Multi-agent System Using Ontological User Profiles for Dynamic User Modelling", in *Proc. of the Web Intelligence and Intelligent Agent Technology, (WI-IAT 2011)*, pp. 430–437, 2011.

[19] J.L. Herlocker, J.A. Konstan, L.G. Terveen and J.T. Riedl, "Evaluating Collaborative Filtering Recommender Systems", *ACM Trans. on Information Systems (TOIS)*, 22(1), pp. 5–53, 2004.

[20] K. Hindriks and D. Tykhonov, in "Opponent Modelling in Automated Multi-Issue Negotiation Using Bayesian Learning", *Proc. of the 7th Int. Joint Conf. on Autonomous Agents And Multiagent Systems - Vol. 1*, pp. 331–338, 2008.

[21] G. Lax and G.M.L. Sarné, "CellTrust: a Reputation Model for C2C Commerce", *Electronic Commerce Research*, 8(4), pp. 193–216, 2008.

[22] G. Lu, J. Lu, S. Yao and Y.J. Yip , "A Review on Computational Trust Models for Multi-agent Systems", *The Open Information Science Journal*, Available: http://dx.doi.org/10.2174/1874947X00902020018, pp. 18–25, 2012.

[23] D. Magalhaes de Sa and L.J. Aguilar, "A Software Agent for Report Development", in *Proc. of the 5th Int. Multi-Conf. of Engineers and Computer Scientists, (IMECS 2008)*, pp. 852–857, 2008.

[24] S. Marsh and P. Briggs, "Examining Trust, Forgiveness and Regret as Computational Concepts", in *Computing with Social Trust*, Ed. J. Golbeck, pp. 9-43, 2009.

[25] C.M. Macal, and M.J. North, "Tutorial on Agent-Based Modelling and Simulation", *Journal of Simulation*, 4(3), pp. 151–162, 2010.

[26] E. Mosqueira-Rey, D. Alonso-Rios, A. Vazquez-Garcia, B. Baldonedo Del Rio, A. Alonso Betanzos and V. Moret-Bonillo, "An Evolutionary Multiagent System for Studying the Usability of Websites", in *Proc. of the Int. Work. on Web-based Tools*, pp. 34–41, 2007.

[27] MUADDIB URL. Available: http:*//muad.altervista.org*, 2011.

[28] I. Pinyol and J. Sabater, "Computational Trust and Reputation Models for Open Multi-Agent Systems: a Review", *Artificial Intelligence Review*, Available: http://dx.doi.org/10.1007/s10462-011-9277-z, pp. 1–25, 2012.

[29] V. Raghavan, P. Bollmann and G.S. Jung, "A Critical Investigation of Recall and Precision as Measures of Retrieval System Performance", *ACM Transactions on Information Systems (TOIS)*, 7(3), pp. 205–229, 1989.

[30] S.D. Ramchurn, D. Huynh and N.R. Jennings, "Trust in Multi-Agent Systems", *Knowledge Engineering Review*, 19(1), pp. 1–25, 2004.

[31] J. Reisinger and R. Miikkulainen, "Evolving Neural Networks", in *Proc. of the 9th Conf. on Genetic and Evolutionary Computation (GECCO'07)*, pp. 3415–3434, 2007.

[32] D. Rosaci, "CILIOS: Connectionist Inductive Learning and Inter-Ontology Similarities for Recommending Information Agents", *Information Systems*, 32(6), 2007.

[33] D. Rosaci D. and G.M.L. Sarné, "MASHA: A Multi-Agent System Handling User and Device Adaptivity of Web Sites", *User Modeling and User-Adapted Interaction*, 16(5), pp. 435–462, 2006.

[34] D. Rosaci D. and G.M.L. Sarné, "EVA: An Evolutionary Approach to Mutual Monitoring of Learning Information Agents", *Applied Artificial Intelligence*, Forthcoming, 2011.

[35] D. Rosaci, G.M.L. Sarnè and S. Garruzzo, "MUADDIB: A Distributed Recommender System Supporting Device Adaptivity", *ACM Transaction on Information Systems (TOIS)*, 27(4), pp. 1-41, 2009.

[36] G. Ruthanna and K. Spears, "You Don't Act Like You Trust Me: Dissociations Between Behavioural and Explicit Measures of Source Credibility Judgement", *The Quarterly Journal of Experimental Psychology*, 65(1), pp. 121–134, 2012.

[37] J. Sabater-Mir and M. Paoulucci, "On Open Representation and Aggregation of Social Evaluation in Computational Trust and Reputation Models", *International Journal of Approximate Reasoning*, 46(3), pp 458–483, 2007

[38] J. Sabater and C. Sierra, "REGRET: A Reputation Model for Gregarious Societies", in *Proc of the 4th Work. on Deception Fraud and Trust in Agent Societies (AGENTS'01)*, pp. 61–70, 2001.

[39] J. Sabater and C. Sierra, “Review on Computational Trust and Reputation Models”, *Artificial Intelligence Review*, 24(1), pp. 33–60, 2005.

[40] Q. Shehory, K. Sycara, P. Chalasani and S. Jha, “Agent Cloning: an Approach to Agent Mobility and Resource Allocation”, *Communications Magazine, IEEE* , 36(7), pp. 58–67, 1998.

[41] D. Srinivasan *Innovations in Multi-Agent Systems and Application 1*, ser. Studies in Computational Intelligence, vol. 310, 2010.

[42] P. Stone, “Learning and Multiagent Reasoning for Autonomous Agents”, in *Proc. of the 20th Int. Joint Conf. on Artificial Intelligence*, pp. 12–30, 2007.

[43] A.M. Uhrmacher and D. Weyns, *Multi-Agent Systems: Simulation and Applications*. CRC Press, Inc., 2009.

[44] C.J. Van Rijsbergen, *Information Retrieval*. Butterworths, 1979.

[45] M. Van Vugt, G. Roberts and C. Hardy, *Competitive Altruism: Development of Reputation-based Cooperation in Groups*, Handbook of Evolutionary Psychology. Oxford: Oxford University Press, 2007.