This is a post-print version of "A Game-Based Competition as Instrument for Teaching Artificial Intelligence".
The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-70169-1_6

A Game-Based Competition as Instrument for Teaching Artificial Intelligence

Federico Chesani[1], Andrea Galassi[1], Paola Mello[1(✉)] and Giada Trisolini[2]

[1] Department of Computer Science and Engineering (DISI), University of Bologna, Bologna, Italy
{federico.chesani,a.galassi,paola.mello}@unibo.it
[2] Department of Education Studies (EDU), University of Bologna, Bologna, Italy
giada.trisolini2@unibo.it

Abstract. This paper reports about teaching Artificial Intelligence (AI) by applying the experiential approach called "learning by doing", where traditional, formal teaching is integrated with a practical activity (a game competition, in our case), that is relevant for AI discipline and allows for an active and playful participation of students.
Students of the course of Fundamentals of AI at the University of Bologna have been challenged (on a voluntary base) to develop an AI software able to play the game of Nine Men's Morris: at the end of the course, the software players have been compared within a tournament, so as to establish the competition winner. The game has been chosen to let the students deepen the knowledge about AI techniques in solving games, and to apply it in a real, not trivial setting.
The significance and the impact of this approach, from the educational point of view, have been assessed through two questionnaires, a first one focused on the technical aspects, and a second one on the students' opinions about the initiative. The results are encouraging: students declare they felt highly motivated in studying AI algorithms and techniques, and they have been stimulated in autonomously search for extensions and new solutions not deeply investigated during traditional lessons.

1 Introduction

Learning how intelligent systems work and are designed is important for new generations which will grow in a world full of "smart objects". New forms of teaching have to be determined to ease people's approach to Artificial Intelligence (AI), not only in the university context, but also in primary and secondary schools, as well as in the industry. Beside traditional teaching techniques, AI should be approached through tasks people are familiar with, such as games, and through practical experiences.

The "Learning by Doing" concept belongs to the pedagogical approach proposed by John Dewey, who founded and supported the so-called "active pedagogy" [4]. In that context, the traditional informative and passive education model was confronted with an active and progressive model based on experiential and participatory learning processes. According to Dewey, education should be considered as an active process where the student can interact with the context, and possibly modify it. While in the past the scientific debate considered traditional learning in contrast with Learning by Doing, nowadays it is highly preferred to merge the traditional informative and passive education model with an active and progressive model based on experiential and participatory learning processes. To facilitate the knowledge building processes, active didactic methods should be considered: as Dewey pointed out in [4], it is necessary to promote practical efficiency with awareness and critical thinking, otherwise the experiential learning is reduced to a routine mechanism.

One of the most effective didactic strategies is represented by Dewey's laboratory classrooms, where students can learn by their direct experiences, put in practice their previous knowledge and, at the same time, acquire new skills. In this theoretical context, students have an important and active role: pupils participate actively in the learning processes and expand their own knowledge and thinking by interacting with the environment through the experience. Teachers become facilitators of the learning process by changing their role from an instructive one towards a collaborative and participative role.

Games and game competitions have played an important role in AI progress: beside providing stimulating research problems, games provide also benchmarks for confronting solutions and approaches. Games are also familiar and well-known to the large public audience, that can easily follow successful results without the need of deep AI knowledge. Moreover, many prestigious universities, such as Stanford and Essex, are exploring games and game competitions as a teaching methodology within their university courses. The University of Huelva (Spain) has made a further step, and has completely integrated the *Google AI Challenge* into its AI courses [3]. These experiments have been successful from many perspectives, resulting in an improvement of students motivation and acquired knowledge.

In [17] the author provides an interesting analysis of AI competitions, and lists a number of characteristics that could lead such competitions to success or to failure. Among the many, we report here the following:

- The development should be independent of the programming language and the operating system.
- The game should be fun for human players.
- The code of the competition and of the past years competitors should be open source.
- The rules of the competition should be clear and stable.
- It should be easy to develop a very simple (and probably not competitive) solution.
- A discussion group for technical support, with both participants and organizers, should be run.

We experimented the principles of the "Learning by Doing" approach in an AI course at the University of Bologna: following the AI traditional fondness for game competitions, we proposed the students to compete on the development of a software agent for the Nine Men's Morris game. Students participated on a voluntary base, and teams (up to four members) were allowed. At the end of the course, the software players have been confronted in a tournament, and students have been invited to give a formal presentation about technical choices (with their colleagues as audience).

The Nine Men's Morris game has been chosen because it is a non-trivial challenge, yet master degree students have the needed knowledge to successfully implement a player. Moreover, the game has been the subject of a number of scientific studies: it has been proved that its solution is a draw [6], a well-performing heuristic function has been calculated with genetic algorithms [11], and a strategy that could lead to better results against a fallible opponent has been defined [7].

The development of an artificial player required the students to apply their AI knowledge into a practical implementation. In particular, students experimented with the application of key concepts of AI game solving such as, for example, Minmax algorithm and Alpha-beta pruning, heuristics search, and optimization. Moreover, students were stimulated to investigate other themes such as machine learning, therefore consolidating and expanding their knowledge. The outcomes of this teaching experience have been evaluated through two questionnaires, one focused on the adopted techniques, and a second one focused on how they perceived the experience. The questionnaires answers confirm the success of the experiment and provide useful indications for further challenges.

The paper is structured as follows. In Sec. 2 we outline the educational approaches which may be relevant to this experience. The competition is described in details in Sec. 3, while Sec. 4 contains the discussion about how the entire experience has been evaluated and which have been the results. Finally, Sec. 5 will conclude the paper.

2 Educational Approaches

The didactic perspective of Constructivism [12] is focused on some principles of "active pedagogy", and in particular on *participation* and *social practices*. Constructivism is an epistemological orientation according to which knowledge is acquired through an active building process, culturally situated and socially negotiated: the student actively participates in building knowledge within a participatory context [10]. The process of learning, strongly influenced by social relationships, is considered to be the result of two factors: the *cooperation with others* (social factor) and the *features of the task* (environmental factor). Therefore, acquiring knowledge is the result of a group interaction: the learning of an individual is the result of a working group that is born from the comparison and collaboration of interdependent groups, and of the use of interpersonal communication methods [9].

In this work the adopted didactic strategies focus on integrating an innovative approach, characterised by experiential and collaborative learning, in a traditional education approach. Within and during the traditional course lessons, the students are invited to join a competition that requires them to apply their acquired knowledge, by working alone or in team. In this laboratorial part of the course, students can put in practice their knowledge with the activation of experiential learning processes.

The laboratorial/practical activity guides the students toward simulating certain hypothesis and verifying them: in an artificial context with conditions similar to reality, students can recognize variables and useful elements required to make a decision. In addition, team working allows the students to discuss, share and create new knowledge, by using social and communication skills and by activating other models such as peer tutoring and reciprocal teaching. Moreover, the collaborative learning approach allows students to access their competences and soft skills which are enhanced by teamwork. This concept was described as the Zone of Proximal Development [18], in which students gradually develop the ability to perform tasks autonomously. This aspect is especially relevant to AI field, where applications typically require diversified skills and interdisciplinary knowledge.

Summing up, the didactic strategy based on gaming can improve students motivation and collaboration. Moreover, its effectiveness can be enhanced by including also the competition dimension in the game.

3 Competition Development

In the following we provide a description of the main aspects of the competition. In particular, we introduce the game of Nine Men's Morris, the educational context of the students and the setting of the competition itself.

3.1 Nine Men's Morris

Nine Men's Morris game (also called Mill or Cowboy Checkers) is a ancient [2] two-player perfect information boardgame. Different versions of the game exist, but in this work we follow the rules described in [6].

Both players have 9 checkers, also called stones or men, of the same colour (usually black for one player and white for the other one). The game board, illustrated in Fig. 1, presents three concentric squares and four segments which links the squares by joining the midpoints of the sides. The 24 points of the board where lines intersects are the position where the players can place their checkers. At the start of the game, the board is empty and the players have their stones in hand. The players alternate their moves, with the "white" player usually making the first move.

The game develops along three different phases which are determined by the number of checkers that players have in their hands and on the board. The rules for each phase are the following:

1. In the first phase, during their turns the players put a stone on an empty position on the board.
2. When all stones are placed, a move consists in sliding a checker along a line into a nearby empty position.

3. When one or both players are left with only three checkers on the board and none in the hand, they can move anyone of their checkers into any empty position on the board.

Players eliminate an opposing checker by "closing a mill", i.e. by aligning three stones along a segment. If there are adversary's stones which are not part of a mill, they must be chosen instead of the ones which are aligned in a mill. To win, players have to remove 7 opposing stones or manages to impede any possible move to the adversary. If a board configuration is repeated, the game is declared a draw (this can happen only during the second and the third phase).

The number of possible board states is included between 7.6×10^{10} and 2.8×10^{11} [6], which is relatively small compared to other classic board games. As a quick comparison, consider that the number of Chess configurations is included between 10^{43} and 10^{50}, while Go complexity has been estimated to be 10^{172} [1].

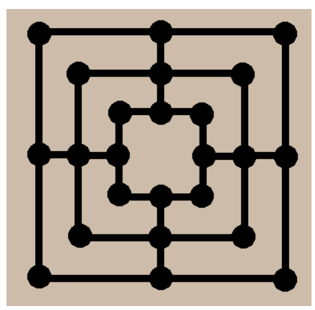

Fig. 1. Nine Men's Morris game board.

3.2 Educational Context

The educational context of the students to whom this challenge is addressed is a fundamental factor to take into account. The usefulness of this experience is strictly correlated with the objectives that the AI course aims to achieve, while its success could be also linked to the competences that students already have.

The Artificial Intelligence Course. The goal of the course is to introduce the main concepts and methods of Artificial Intelligence, with a particular emphasis on symbolic systems and problem solving, knowledge representation and reasoning, and computational logic.

The problem solving topic, intended as a search in a state space, is introduced along with the main search strategies. Games, constraint satisfaction problems, and planning problems are chosen as practical instances and deeply discussed in the course both from theoretical and from practical points of view. Regarding games in particular, Minimax-Decision algorithm and Alpha-beta search are introduced, together with some examples related to a number of games. The adopted reference book within the course is [14]. During the course, the design and the implementation of some practical AI systems based on procedural and declarative programming languages (Prolog in particular) are also stressed. Moreover, seminars on specific AI topics are planned: to cite few, an introduction to genetic algorithms, and an overview of Natural Language Processing and Understanding.

The course is the first experience about AI for the Master students in Computer Engineering, and is intended as preparatory course for the Intelligent Systems course, where Machine Learning techniques are introduced together with applications of problem solving techniques in scheduling, planning and optimization. Therefore, no strong prerequisites are required: the student is gradually introduced to the fundamental notions of the AI. However, basic notions of a high level programming language are required, to successfully understand case studies and applications presented during the lessons.

Classroom lessons are interleaved by some practical activities in the laboratories. For example, an activity focuses on the use of the software library AIMA[3], a collection of AI algorithms implemented in different programming languages such as Java, Lisp and Python. Moreover, autonomous AI projects are welcomed and promoted by suggesting ideas and possible case studies. Further details about the course are available on the university website[4].

Bachelor Degree Background. Most of the students who attend the Artificial Intelligence course hold a bachelor degree in Computer Engineering from the University of Bologna. Hence, the majority of the students already have a background knowledge about object oriented programming and about principles and patterns of software engineering. Moreover, students who attended the bachelor degree are also proficient in topics like concurrent programming, computer networks, and information systems and databases. The computer science-related courses in the bachelor degree comprise traditional lessons (held in classrooms), as well as laboratory sessions, where students reinforce the acquired knowledge and apply it on practical experiences, for example by programming in languages such as C, Java and C#.

3.3 Competition Description

The competition is presented to the students during the course lessons, usually in the last weeks of March or the first ones of April. Students are required to "register" for the competition within one month from the presentation in the course, and they are allowed to withdraw at any moment. Such freedom in

[3] http://aima.cs.berkeley.edu/code.html

[4] http://www.engineeringarchitecture.unibo.it/en/programmes/course-unit-catalogue/course-unit/2016/385372

Table 1. Team composition

Group members	♯ groups
1	3
2	10
3	8
4	4

the competition registration/withdraw is meant to avoid adding an additional stress factor to the students, and instead propose the competition as a further opportunity. To encourage the participation, a small reward is granted on the final course grade: a bonus of 2 points out of 30, which means up to 6.67% of the final grade. To encourage teamwork without imposing limitations (some students might prefer to work alone), a competition team can be made of 1 up to 4 students. The deadline for the project submission is usually at the end of May, after which students are required to prepare a presentation that will be given to the classmates. The total time available for working on the project is therefore of 8-10 weeks, plus one additional week for the presentations. Students are strongly encouraged to communicate with the competitors, and to share knowledge between the teams: moreover, the presentation given in front of the class mates serves as a further chance for discussion and confrontation.

The software structure of the competition is quite simple: a Java program acts as referee of the game, while two other programs act as players. The referee communicates to the players with TCP protocol, sending them the state of the game in the form of a Java object and asking them the move they intend to make in the form of a textual string. Students are provided with the referee software, along with some classes representing the problem, the game state, as well as an example player (which is simply a text-based interface for playing). The AI software player should be capable of playing the game, i.e. to read the game state provided by the referee, and to send back the chosen move. For each turn the players have a maximum time limit to provide the move to the referee (currently, 60 seconds), otherwise they are declared defeated.

The competition is organized as a round-robin tournament: each player has to play against all the other players twice, once with white checkers and once with blacks. When all the matches have been played, the final result of each match is published on the website of the course, along with the winner of the competition. Records of the matches, along with a brief description of the players and their compiled code is maintained on the competition website[5]. The competition is run by academic staff in a controlled environment for sake of fairness.

The competition has been held for three years consecutively (academic years 2014/2015, 2015/2016, and 2016/2017). The number of participants has been 17, 27 and 19, respectively, while the number of teams has been 7, 10 and 8. Table 1 shows the number of teams w.r.t the number of participants: clearly, students generally preferred to work in teams.

[5] http://ai.unibo.it/mulinochallenge

4 Educational Experience Assessment

In this Section we introduce and discuss the results of the competition. Students were asked to answer two questionnaires. In the qualitative questionnaire students provided their feedback on a number of different aspects related to the usefulness, impact, and organization of the competition. The technical questionnaire is focused more on the technologies adopted by the participating teams. The former questionnaire aims to capture the perception of the students about the competition, while the latter one aims to spot AI techniques and solutions that were not covered in the course but used anyway within the competition, thus pointing towards a certain degree of autonomy and maturity of the students. We have also compared the exam marks of students who have participated to the challenge and students who have not.

4.1 Evaluation of Students Opinions

At the end of the competition, the participating students have been asked to answer a qualitative questionnaire of 20 questions, 18 of which are based on a five-level Likert scale (1 = strongly disagree; 5 = strongly agree), one question requires a numeric estimate and one question simply asks for suggestions. Table 2 shows the questions, that have been strongly inspired by the questionnaires used in [3]. Figure 2 reports average and standard deviation of the answers to the 18 questions based on the Likert scale.

The questionnaire is divided into the aspects of interest for this study: (*i*) if the experience has enhanced or increased the students' knowledge (Q1-Q5); (*ii*) if the competition has stimulated the students (Q6-Q10); (*iii*) if it has enhanced skills which are not strictly related to AI (Q11-Q15); and (*iv*) how much it has been difficult to take part in the competition (Q16-Q19). Answering the questionnaire was not mandatory, only 36 out of 63 students filled it.

Students agree that this experience has improved their knowledge about AI concepts (both new and already known) and problem solving, while are neutral about how it has influenced their programming languages knowledge. Students' motivation has resulted to be stimulated by competing in a challenge, indeed their feelings about the course have been improved by this experience, even though they were already high. Students' skills have been positively influenced by the competition and their perception of the value of sharing information has increased. Despite the mixed feelings about the difficulty of the experience and the amount of hours spent for it, students agree that this challenge has been well organized and, more importantly, has been a satisfying experience.

It is worth noting the quite important standard deviation of the amount of hours that students have dedicated to the project. A possible explanation would take into account the number of members in a team: it is likely that a student participating alone will spend more time on the development of the software than a student who is part of a four members team.

Table 2. Student's quality questionnaire

#	*Knowledge*	*Avg*	*SD*
1	The experience enables the consolidation of theoretical concepts on AI	4.36	±0.63
2	The experience allows new concepts on AI to be acquired	4.42	±0.76
3	The experience allows to discover new ways to solve problems	4.17	±0.73
4	The experience allows new concepts on language programming to be acquired	3.14	±1.11
5	My ability to apply knowledge in practical and real problems after the challenge is positive	4.17	±0.83
#	*Interest/Motivation*	*Avg*	*SD*
6	My general assessment for the course before the experience is positive	4.11	±0.74
7	My general assessment for the course after the experience is positive	4.47	±0.60
8	The result obtained in the challenge influences learning on AI	3.69	±1.05
9	The result obtained in the challenge influences interest and motivation	3.97	±1.09
10	Competing in a challenge promotes motivation and interest	4.58	±0.72
#	*Personal Skills*	*Avg*	*SD*
11	The value of sharing information before the challenge is positive	4.00	±0.88
12	The value of sharing information after the challenge is positive	4.19	±0.70
13	The challenge serves to better understand personal skills	3.97	±0.93
14	The experience allows knowledge on work organization to be acquired	3.83	±0.93
15	The experience allows knowledge on cooperation and teamwork to be acquired	4.19	±0.91
#	*Workload/Difficulty*	*Avg*	*SD*
16	The difficulty and workload of this practice/experience is high	3.36	±1.13
17	How many hours have you spent working on this project? [a]	38.41	±25.09
		38.83	±24.11
18	The general assessment on development and organization of this practice is positive	4.06	±0.74
19	My general assessment for this practice/experience is positive	4.31	±0.81
#	*Suggestions*		
20	How could this experience be improved? [b]		

[a] Five answers, such as "Enough", did not express a quantifiable amount of time, therefore they have not be taken into account. For 3 answers which indicated an interval, such as "20-30", we have considered the mean value of the interval.
Four answers used different time units, such as days. We have not considered these answers in the first line. In the second line we have considered them estimating 4 hours of work for each day.

[b] Obviously this question does not have associated values.

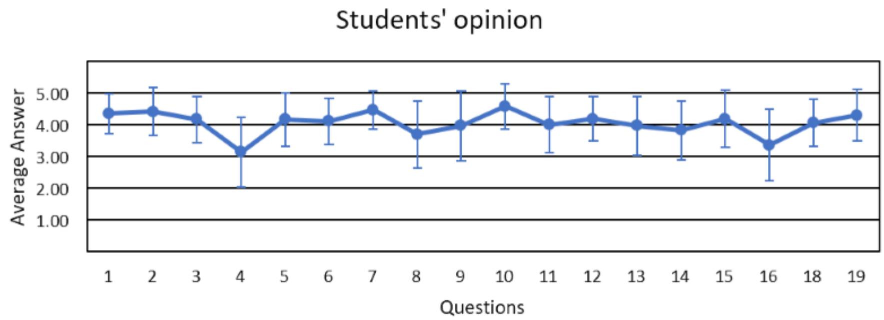

Fig. 2. Average scores and standard deviation for the students' opinion based on a five-level Linkert scale

Students' Suggestions. From what has emerged from the suggestions, the main lack of our competition had been the absence of a "language agnostic" referee. The choice to have a referee that communicated the states through a Java object seems to have, understandably, made the participants feel forced (or at least encouraged) to use Java as development language.

Another strong desire has been to have results of the concluded matches immediately available, not only at the end of the competition, thus indicating that the competitive component is important for the students.

One last popular request has been about a greater amount of time for developing the project, and for studying and developing better AI techniques which are not taught during the course, for example machine learning algorithms.

Some minor suggestion have been the following: to use a less known game as benchmark, to dedicate one laboratory lesson to the development of an artificial player, to give a special reward for the winner of the competition and to run the competition on a hardware equipped with a modern graphic card, so as to exploit it for, e.g., training deep networks [8].

Besides these helpful suggestions, some students have expressed a deep appreciation toward the competition.

4.2 Evaluation of the Used Techniques

To collect information about the techniques used by the students, two instruments have been used. The first one is a mandatory presentation of the project that each work group made to illustrate its software: the audience comprised the course teachers and the students. The second one is a questionnaire with multiple choices and open questions. As the quality questionnaire, its filling was completely volunteer, therefore available information regards only 24 teams out of 25.

The most frequently chosen programming language is Java, and only one team opted to develop the project in a different language, choosing C. Almost every team has used traditional state-space search as AI technique, except one team which has chosen to apply neuroevolution, in particular the NEAT algorithm [16]. Genetic algorithms were also exploited by a second team, which has used them to tune the parameters of the heuristic function used for the search. As search algorithm, most of the teams have used Iterative Deepening with A* or MiniMax with Alpha-Beta pruning. Surprisingly, seven teams have studied and applied other algorithms which are not covered by the university course, namely Negamax, NegaScout [13] and Best Node Search [15]. Four teams have also exploited part of the problem symmetries and transposition tables to speed-up the search. Many students have chosen to develop the whole software code by themselves; nonetheless, 10 teams relied on libraries such as AIMA and minimax4j[6].

As emerged from the questionnaire and from the presentations, even though most of the teams have implemented techniques which have been taught during the university course, some teams have demonstrated spirit of entrepreneurship: students have autonomously studied and implemented techniques which were new for them.

The software agents which, so far, have obtained the best results, rely on Iterative Deepening, compact and easily accessible representation of the game state and symmetries consideration to reduce spatial search.

4.3 Comparison between exam marks

For the sake of completeness, we have also analyzed how the students have performed in the course exam. We have taken into account the results obtained between June 2015 and July 2017, for a total of 246 exams. We have computed the average mark scored by students and the relative standard deviation, considering only sufficient marks (above 18/30) and only the most recent mark for each student[7], for a total amount of 228 tests. The "30 cum laude" mark has been counted as 31. The average mark of students who have participated to the competition is 28.22, while students who did not participate got an average mark of 27.55. The standard deviation for the former group is 2.76, while the one for the latter group is 3.18. On average, students who have taken part to the initiative have obtained a mark of 0.67 points higher than the other students. Moreover, considering all the exams performed during the same time span, the percentage of failed exams results to be 14% for students who have not taken part to the competition and 5% for students who have.

Apparently, students who took part to the competition witness better results, in terms of final marks , and in terms of less failed tests. Even if this result seems to suggest that the initiative has been effective, there are other aspects to be considered as well. First of all, it is debatable if students advancements can be testified only by the final marks. For example, it might be worthy to look at the number of time that students have repeated the exam and their previous results,

[6] https://github.com/avianey/minimax4j

[7] Students are allowed to repeat the test, even in case of sufficient marks.

the time occurred between the end of the lessons and the exam, and whether the students have personally attended the course lessons or not. Second, it is not possible to determine whether the competition has improved the students' performance or the participants were already the most skilful and motivated students: in the latter case, collected data would simply show that motivated students are more prone to take part to side initiatives such as our challenge.

5 Conclusions and Discussion

On the basis of the results emerged from the questionnaires, we deem this experience successful. Students confirmed the improvement of their knowledge and stated an increased interest toward the course topics and AI in general. The effectiveness of the adopted learning approaches is confirmed as well.

According to [17], and also from what has emerged from the students suggestions, an important lack is related to not making the competition sufficiently independent of the developing language: many students felt "obliged" to use the Java language. The amount of hours dedicated by students to this experience is highly variable and some of them have expressed the need for more time. This could be possible only by extending the competition beyond the course end. This possibility has been taken into account but it has been discarded due to the community environment that the course offers to the students. Often students took advantage of pauses between lessons to discuss about the competition both with professors and members of other teams. We think that this opportunity is a fundamental part of the competition, because it can be interpreted as an informal discussion group, another important element of success suggested by [17]. One last important aspect to underline is that the enthusiasm towards this experience has led many students to develop game-related university projects and master thesis, which may led to further research works, such as genetic algorithms or deep networks applications to the game context [5].

This experience can be extended to learn other AI techniques such as machine learning, natural language processing, constraints solving optimization, planning and scheduling, and computational logic. Also, it would be interesting to investigate other AI platforms developed in academic contexts or industrial settings, and to determine how suitable these platforms would be for creating similar experiences and for improving students learning.

Acknowledgements

We would like to thank all the students who have participated to the competition, and whose names are available on the competition website. Moreover, we are especially grateful to those students who answered our questionnaires. We would like to thank also the reviewers for their helpful and detailed suggestions.

References

1. Allis, L.V., et al.: Searching for solutions in games and artificial intelligence. Ponsen & Looijen (1994)
2. Bell, R.C.: Board and table games from many civilizations, vol. 1. Courier Corporation (1979)
3. Carpio Caada, J., Mateo Sanguino, T., Merelo Guervs, J., Rivas Santos, V.: Open classroom: enhancing student achievement on artificial intelligence through an international online competition. Journal of Computer Assisted Learning 31(1), 14–31 (2015), `http://dx.doi.org/10.1111/jcal.12075`
4. Dewey, J.: How we think. DC Heath (1910)
5. Galassi, A.: Symbolic versus sub-symbolic approaches: a case study on training Deep Networks to play Nine Men's Morris game. Master's thesis, University of Bologna (Italy) (Mar 2017), `http://amslaurea.unibo.it/12859/`
6. Gasser, R.: Solving nine men's morris. Computational Intelligence 12(1), 24–41 (1996), `http://dx.doi.org/10.1111/j.1467-8640.1996.tb00251.x`
7. Gévay, G.E., Danner, G.: Calculating ultrastrong and extended solutions for Nine Men's Morris, Morabaraba, and Lasker Morris. IEEE Transactions on Computational Intelligence and AI in Games 8(3), 256–267 (Sept 2016)
8. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), `http://www.deeplearningbook.org`
9. Kaye, A.: Learning together apart. In: Collaborative learning through computer conferencing, pp. 1–24. Springer (1992)
10. Lave, J., Wenger, E.: Situated learning: Legitimate peripheral participation. Cambridge university press (1991)
11. Petcu, S.A., Holban, S.: Nine men's morris: Evaluation functions. International Conferencs on Development and Application Systems pp. 89–92 (2008)
12. Pumilia-Gnarini, P.M.: Handbook of Research on Didactic Strategies and Technologies for Education: Incorporating Advancements: Incorporating Advancements. IGI Global (2012)
13. Reinefeld, A.: Spielbaum-Suchverfahren, vol. 200. Springer-Verlag (2013)
14. Russell, S.J., Norvig, P.: Artificial intelligence (a modern approach) (2010)
15. Rutko, D.: Fuzzified algorithm for game tree search. In: Second Brazilian Workshop of the Game Theory Society, BWGT (2010)
16. Stanley, K.O., Miikkulainen, R.: Evolving neural networks through augmenting topologies. Evolutionary Computation 10(2), 99–127 (2002), `http://dx.doi.org/10.1162/106365602320169811`
17. Togelius, J.: How to run a successful game-based ai competition. IEEE Transactions on Computational Intelligence and AI in Games 8(1), 95–100 (March 2016)
18. Vygotskij, L.S.: Myslenie i rec psichologiceskie issledovanija. Movska-Leningrad: Gosudarstvennoe Socialno-Ekonomiceskoe Izdatelstvo [It. tr. Pensiero e linguaggio. Bari: Laterza, 1992] (1934)