Machine assisted problem solving;

Cooperative problem solving with an IDSS assisting humans

Oskar Johansson, Justin Greep, Axel Wickman, Amanda Rombo, Jenny Andersson, Róbert Snaer Hilduson, Hannes Karlsson

Abstract

Research on problem solving has been extensively performed related to several different aspects of problem solving. The present study was conducted with the purpose to investigate whether it was possible to build an intelligent decision support system (IDSS) that could be helpful in aiding humans in problem solving compared to humans alone. The IDSS was implemented using a Q-learning algorithm trained by a custom-built bot. The problem solving was tested in an environment where three separate problems were implemented in a 3D-spaceship environment. Participants were tested with and without IDSS-assistance using a between-group design. The results generated could not be used to draw any significant conclusions regarding the problem formulation of the problem.

Introduction

Research conducted about problem solving is not an unusual finding in studies within cognitive sciences. This is due to the importance of problem solving abilities on how well we can act and adapt to the world around us (Jonassen, 2000). Humans possess many important built-in mechanisms related to effectiveness in problem solving – for example the ability to notice the most important aspects of a problem, find similarities related to previous knowledge and a sensitivity to the context in different situations (Duris, 2018). This altogether accounts for the phenomena that humans alone are generally better at problem solving than computers. The purpose of the present study was to evaluate and compare how well inexperienced humans solved problems in a virtual 3D-environment compared to the same procedure but with the supports of an intelligent decision support system (IDSS), implemented with the help of a Q-learning algorithm. Worth noting is that the original objective of this study was to implement the IDSS with the help of a partially observable Markov decision process (POMDP), however this was left out due to the high complexity of this implementation which did not fit into the time scopes of the project.

Delimitation for the present study include limiting testing to students at Linköping University, implementing a menu for communication with the IDSS instead of using Natural Language Processing (NLP) as well as using only time to measure participants problem solving performance.

Theory

IDSSs are used for computer assisted decision making where high risk decision making is critical for success (Philips-Wren, 2012). Most commonly, it is implemented using some form of an artificial intelligence that provide relevant information from a data set where human would have trouble extracting the most relevant information related to the aims of the situation. The limitations of analytic ability in human decision makers is a motivation for why an IDSS would be a possibly valuable tool for assisting inexperienced human in a problem solving environment. The IDSS was in the present study implemented by using a Q-learning algorithm. Q-learning is an optimal strategy for learning agents in a discrete world, using a model-free approach. The agent will be rewarded in different degrees depending on the value – payoff – of a certain action in a certain state. The overall goal for the agent is to maximize the payoffs, resulting in a form of reinforcement learning where the agent learns from experience.

The three problems that were implemented in the environment were a Tower of Hanoi-

variation, a light switch-problem and a maze. Jonassen (2000) defines a problem as containing two main attributes; the first is that a problem is said to be an unknown difference between a goal state and a current state for a given situation. The second is that solving this unknown entity must have some social, cultural or intellectual finding – someone must find the solution worth finding.

The program

To investigate the aspects of the problem formulations we were required to develop a program for using when testing participants and that could be used both with and without the IDSS-assistance. The contents of this program includes a 3D-environment where the problems are implemented, the IDSS build with the Q-learning algorithm and a bot used for training the algorithm.

Environment

The testing was set in a spaceship-environment containing three problems for the participants to solve. These three problems were thought to test different aspects of problem solving, for example mentioned by Duris (2018). The cross-platform game engine *Unity 3D* was used for building the environment. A spaceship was specifically chosen because of the high customizability factor and the possibility to implement unreal implementations of the different problem than could have been possible in a simulated real environment. The spaceship consisted of a three-way corridor which was connected to two different rooms containing the Tower of Hanoi and the light switch problem. When the first two problems were both finalized, a door opened to the maze.

The Tower of Hanoi was implemented using a classic move-problem variation (Hayes & Simon, 1974) containing three disks. To contextualize this problem, it was described to the participants as batteries being moved from a charging station to discharging station. This problem was implemented to test cognitive mechanisms related to planning and pattern recognition.

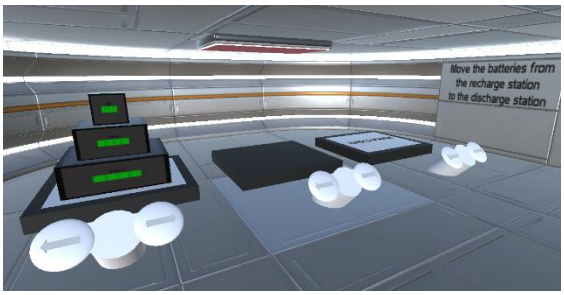

Fig 1. Overview of the Tower of Hanoi-problem

In the light switch-problem, the participant stands in front of a wall with different-colored buttons, a light switch and a reset-button. The problem was described as the buttons having to be pressed in the right order to restart the power supply. This problem was inspired by the switchlight problem introduced by (Speedie, Trefflinger & Houtz, 1976). However, instructions were needed to be implemented to make the results dependent on something else than the participants only guessing the correct combination. Implementing instructions that were somewhat confusing was assumed to cause a need for the participants assistance of the IDSS. This was important since instructions that were too easy would require no need of the IDSS in any situation. This problem was implemented to test cognitive mechanisms related to pattern recognition and working memory.

Fig 2. Overview of the light switch problem

The maze was implemented in a random spatial system and was presented when the

first two problems were finalized by the participant. There were multiple routes to take to solve the maze which meant introducing chance to which route (longer or shorter) the participants decided on. This problem was implemented to test cognitive mechanisms related to spatial knowledge and planning.

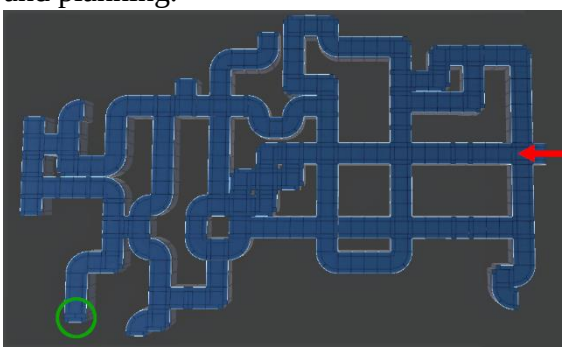

Fig 3. Overview of the maze

When designing the environment, it was required that the participant would understand how to interact with, and navigate though, their surroundings. It is important to take into consideration the cognitive mechanisms used for learning and processing information that will be important factors in the success and understanding of the participant. Smith and Ragan (2005) provides a detailed framework for instructional strategies relating to environmental design. They mention several aspects, called events. The events that were found to be most relevant for the objectives include: gaining attention, information participant of objectives, presenting stimulus materials and providing feedback from the environment. Specific decisions made in order with these events for the final environmental design include; big instructions on what is the goal for the participant written on the walls of the environment, colorful buttons and feedback provided using for example auditory responses when buttons were pressed.

Bot

Since most reinforcement learning algorithms require large amounts of data in order to be able to learn good strategies it is very time consuming to collect data. This presented a problem for the project which was solved by building a talkative training bot (TTB) that could be used for training the IDSS (Q-learning algorithm). The TTB is an autonomous agent able to solve problems in novel environments and has the ability to formulate questions to the IDSS. This means that the reward for the TTB also could be used to reward the IDSS. To mirror human behavior the TTB used a tweaked version of a reinforcement learning algorithm which required no training prior to the introduction of the environment. The implemented algorithm makes it able to solve different kinds of problem. An understanding of hidden states was required for the light switch-problem, where environment feedback is not directly yielded until a right or wrong combination is entered. For the Tower of Hanoi-problem, a way to input human intuitions about what each button does and how the states transition were needed for planning along with an ability to avoid local optima. For the maze problem, the bot needed to be able to consider different positions and orientations to assume in order to explore the map in order avoid a too systematic search approach, part of which meant allowing the bot to forget parts of the map where it had not been in a while.

IDSS

The usage of the IDSS in the environment happens in three steps; data collection, training, and utilization. When running the IDSS, it opens up a WebSocket server which can be used via internet if the proper network socket is opened, or locally. The client tells the IDSS if it should be training, the model name to be loaded and saved to, how often the model should be saved, the observation file name (if any) that should be loaded and saved to, the number statements and observations that can be made by the

user, and the number of possible responses available. During data collection, the client sends all observations and statements in the form of their indexes and reward, and the IDSS logs this to the observation file along with the response that is sent back and then shown on screen. Here, this response is chosen randomly. After the data collection is done, the client tells the IDSS to go into training phase where all the data in the observation file is used to train a Q-table using the Bellman equation (Russel & Norvig, 2014) by looping through all of it in epochs. This Q-table is then saved as a model to a file, which can then be loaded to utilize the IDSS. It should be noted that this system in the system in the future easily could be expanded with more sophisticated reinforcement learning algorithms, like a POMDP.

We used a standard Q-learning algorithm. The learning rate and discount factor was constant during training with values of 0.2 and 0.75, which was chosen through testing. In total, 80 runs of data were collected from the TTB, and was trained on about 60000 times. When choosing an optimal action using the Q-table, no randomness is involved, and a 1-step lookahead.

The IDSS needed to acquire information about the user's current situation to be able to give useful directions. It was first considered to use natural language processing (NLP) for this so that the participants simply could ask the IDSS for assistance out loud, but the form of communication that was decided on for this purpose was a Radial Communications Menu (RCM). The reason for this was that the error rate for NLP along with the infinite number of things that could have been asked if participants could ask whatever they wanted would make this implementation too time-consuming and prone to errors. The RCM-menu consisted of two main buttons in the middle of the screen forming a circle when combined. The menu could then be further expanded to a new layer of buttons, giving the participant difference choices on which information to provide to the IDSS depending on their present situation. Samp and Decker (2010) showed that mouse pointing times were highest in menus like the RCM when compared to other menus. The visual search times were less optimal since time was used to measure performance and compare the two situations, but a quick training session was given to the participants to compensate for that.

Method

In order to test participants both with and without the assistance of the IDSS to get results that could be compared to each other, two tests were performed called situation 1 (without IDSS) and situation 2 (with IDSS). Participants were recruited using convenience sampling and a between-group design was used. The choice of design was motivated by the difficulty of reducing learning effects in an in-group design, making the IDSS irrelevant if the participants already knew the solution for the problems after having tested them without assistance first. This also applies to the possibility of quicker times in situation 1 for participants that had started with situation 2.

The test began by the examiner reading a manuscript with instructions to the participant. Then the participant was asked to give consent and answer a short survey which also assigned them an anonymous test-ID. Then, the test started in situation 1 and in situation 2, a short training session for the RCM was conducted. The participants had 30 minutes to solve the problems. When the participants were finished, they were asked to fill in another survey regarding their test-experience and confidence in solving the problems. Comparison and evaluation between the

two situations were made by comparing the time for completing each individual problem as well as the total test time.

Efforts were made to keep the conditions between the two situations as similar as possible, however, some differences in the nature of the situations required differences in the testing as well. For example, participants in situation 2 had to have a training session with the RCM as well as answer some questions in the after-survey relating to the experience with the IDSS assistance.

Results

Results from both situations were gathered and included total time, time spent per problem and results from the surveys performed by the participants before and after the test.

The general summarization of the results is that the testing with the IDSS did not work as hypothesized. Instead of helping the participants when they asked for help, it gave answers that were unrelated to the problem at hand, or simply a wrong answer. This happened enough times to confuse the participants, making them get stuck on mainly the light switch-problem, which had the highest total time as well as the lowest confidence rate. No significant values were found when performing an unpaired t-test on the individual times and the total time.

Discussion

The main reason for the IDSSs inability to aid the participants was assumed to be dur to the lack of memory in the Q-learning algorithm. For example, 10 possible statements from the user meant 10 states in the algorithm with only one optimal option. This means that without remembering something about the participants whereabouts or previously asked questions, the IDSS could rarely know which problem the participants were facing, not which answer was the correct one. This caused further problems relating to the way the participants acted in situation 2. Based on observations from the examiners that supervised the participants it was noticed that several of the participants always did what the IDSS told them to, and not realizing the instructions were wrong. This was of course misleading, but some of the participants instead chose to discard the IDSS when they noticed that it was not helping them. This is an interesting observation for further research – some of the participants that trusted the IDSS "too much" ended up not finalizing the test within the set time at all since they got stuck in a thought loop in the light switch-problem. This was assumed to happen as a result of the combination of the unclear instructions for the problem itself and the wrong guidance from the IDSS.

Another thing that is important to note is that since we are measuring the performance in terms of time – the time it took for the participants to interact with the RCM in situation 2 can be assumed to have participated to the non-significance between the times in the different situations. One way to counteract this effect would have been to look at the menu times for situation 2 and calculated the times into how long it roughly would have taken to ask the questions in speech instead to see if any other results could have been found if a natural language processing-system would have been used for communication with the IDSS instead of the RCM.

There were a lot of other aspects that could have affected the results from the testing, but none that were regarded nearly as relevant as the problems with the IDSS. These would have been more interesting to discuss if the results would have been more significant. For example, the participants had trouble noticing that you had to stand

rather close to the interactive parts of the environment to be able to interact with them, for example clicking a button. Also, the relatively low number of participants included in the study decreases the validity and reliability of the results. The choice of a spaceship-environment that was implemented in a game-like way could also have been resulting in participants with much experience of computer-games being better at the problem solving only because they felt comfortable in interacting with the environment.

In further research, it would be interesting to follow one of the two original objectives of the study – using NLP instead of a communication menu and/or using a POMDP instead of a Q-learning algorithm in the IDSS. Another interesting aspect would be to focus more on the misplaced trust in the IDSS that turned out to be important to the participants problem solving efficiency. Research regarding the psychological and cognitive aspects of being given incomplete or wrong information when you are already having issues with solving a problem would be interesting to conduct, possibly in order to find mechanisms that would help participants not getting stuck in thought loops and applying the results to more realistic problems and environments.

Conclusion

In conclusion, the present study was not able to show a significant difference between the two situations (with IDSS and without) adhering to the hypothesis, nor between their confidence in problem solving. However, some aspects found in the developments of this program could be regarded as interesting for further research.

Acknowledgements

We would like to thank our supervisors Arne Jönsson and Erik Marsja for helping us in the development of this project.

References

Duris, F. (2018). “Arguments for the effectiveness of human problem solving”. In: *Biologically inspired cognitive architectures* 24, pp. 31–34.

Hayes, J.R and A.S Herbert (1974). “Understanding written problem instructions.”

Jonassen, D. H (2000). “Toward a design theory of problem solving”. In: *Educational technology research and development* 48.4, pp. 63–85.

Phillips-Wren, G. (2012). “AI tools in decision making support systems: a review”. In: *International Journal on Artificial Intelligence Tools* 21.02, p. 124.

Russel, S and P Norvig (2014).´Artificial Intelligence: A Modern Approach, pp. 873–900.

Samp, K. and S. Decker (2010). “Supporting menu design with radial layouts.” In: *AVI. Cite seer*, pp. 155–162.

Smith, PL and TJ Ragan (2005). “A framework for instructional strategy design”. In: *Instructional design*, pp. 127–150.

Speedie, S. M., D. J. Treffinger, and J. C. Houtz (1976). “Classification and Evaluation of Problem-Solving Tasks. (English”. In: *Contemporary Educational Psychology* 1.1, pp. 52–75.