Research Review

Knowledge Representation Using Markov Decision Process (MDP) for Intelligent Decision-Making in Student Advising Systems

Knowledge Representation final exam , Miss Khawla Tadist , university Ueuromed of Fez , AI Department , FEZ

Presented by
EL HARRARI ANAS & ABDALLAOUI Mohamed
Department of Artificial Intelligence, Ueuromed University
anas.elharrari@eidia.ueuromed.org
mohamed.abdallaoui@eidia.ueuromed.org

ABSTRACT

Student advising systems are tools and methods used by educational institutions to advise and guide students throughout their academic careers. These systems attempt to give tailored support, tools, and counselling to students in order to help them make educated decisions about their educational and professional aspirations. This type of system become very popular and constitutes a great domain of interest as it is a good solution to develop the quality of education and facilitate academic analyse and processing. Student advising systems aim to improve student assistance and enable students to make educated decisions about their academic journey, while also encouraging timely degree completion and student success. Several approaches were used by researchers to develop this type of system which present a lot of interest. However, various challenges remain facing these systems which make this field an interesting subject of research and studies.

Problem Introduction

This study aims to explore the potential of the Markov Decision Process (MDP) in student advising systems. By utilizing MDPs instead of traditional decision trees, these systems can better represent information and make more informed decisions. The ultimate goal is to enhance the ability of these systems to provide personalized and individualized recommendations that are tailored to each student's unique needs. Through this research, we hope to improve student outcomes and promote academic success.

Nowadays, a lot of researchers are using the decision tree algorithm to develop a student advising system. Decision trees are a popular machine-learning technique that uses a tree-like model of decisions and their possible consequences. However, this methodology is usually facing various limitations: this algorithm oversimplifies complicated decision-making processes, risking missing critical subtleties and interdependencies. Furthermore, decision trees are not adaptable, may not completely comprehend the contextual aspects surrounding a student's academic path, and can become difficult to read as they get more complicated. Also, the quality and representativeness of the data used to train the system can have a significant impact on the effectiveness of the system's suggestions, potentially introducing biases or restrictions. Finally, decision tree-based algorithms lack the human aspect and intuition that might be critical in student counselling.

In this research, we will be aiming to develop an MDP-based student advising system instead of the traditional decision tree approach and explore the main advantages and benefits of this methodology.

Objectives

By investigating the possibilities of the Markov Decision Process (MDP) in student advising systems, this research project seeks to make a substantial contribution to the field of educational research. The project will focus on several key goals, including the development of an MDP-based knowledge representation framework for student advising systems, a comparison of the benefits of MDPs over conventional decision trees in terms of decision optimality and personalized recommendations, and a review of the effects of MDP-based knowledge representation on student satisfaction and academic achievement. Furthermore, the research will strive to obtain insight into the benefits, drawbacks, and potential enhancements of MDPs in student advising systems.

This study intends to increase understanding of intelligent decision-making and knowledge representation in the context of student advising through a well-structured article, eventually leading to the creation of more effective and individualized educational systems.

Research methodology

Conducting a literature review, collecting data, designing an MDP-based framework for student advising systems, implementing and testing the framework, comparing it to conventional decision trees, and documenting the findings in a research paper to contribute to the knowledge representation and student advising domains are all part of the methodology for this research.

Literature Review

An MDP is a mathematical tool for modelling decision-making issues in a probabilistic setting. It is a structured depiction of a sequential decision-making process in which an agent interacts with its surroundings.

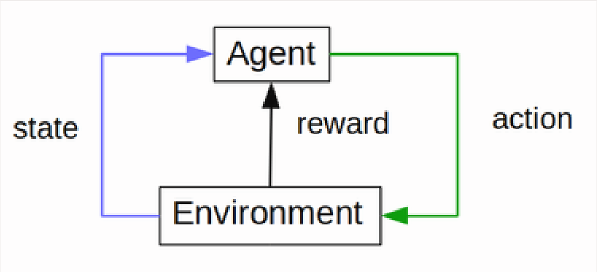
The use of MDP in student advising systems has various advantages, including structured decision-making, flexibility, scalability, and a data-driven approach. MDP can handle uncertainty and maximize long-term results for pupils by exploiting probabilistic transition probabilities. It also allows for customisation and adapts to specific student needs while handling complicated decision-making circumstances effectively. MDP may use past data and student input to continually enhance advising quality and adapt to changing student requirements. Overall, MDP can improve academic performance and student happiness, making it an important tool for improving the efficacy and efficiency of student advising systems.

Data Collection

To simulate our implementation and for a better analysis of our system results, we choose specific data that reflect a good representation of needed information and match the critics of our system structures. The data is created, by our team, for specific reasons related to the research work (simulation, structure representation...) and represent the maximum student profiles, courses, and prior academic achievement.

MDP Framework Design

In order to develop our advising student system using the MDP framework, we followed a specific design that, at the same time, respects the general MDP approach and reflects au maximum of our research interests. This design includes mainly: *State-space definition* which includes various variables that reflect the student's academic context such as the student's academic progress, current year, major...etc.*The action space* basically represents the action that the student may take or do inside the advising system(taking a course, adjusting schedule...etc). *The reward function* quantifies the desirability between each state-and action. This reward function is calculated essentially using the academic performance and the completion of a requirement.



One o the essential parts of MDP design which is used also in our implementation is to define the transition probabilities which refers to calculating the probability of passing from one state to another based on some specific parameters. In our research, we use historical data to decide this transition based on the action taken.

We may remind that one of the main purposes of using the MDP architecture over the traditional tree-based systems is the capacity to keep up with the environment changement and the adaptation to the uncertainty context which constitute a good interest of the proposed research.

TECHNICAL APPROACH

In this research implementation, all the code is developed using the C++ language. The vision behind choosing C++ remains in its capacity for memory manipulation and its great power regarding the representation and the stoking of data in the memory. C++ language is one of the most suitable languages for Knowledge representation systems and data structures problems in general.

It is also important to notice that the presented implementation is essentially a system structure description and a simulation with chosen data (which does not include an interface or GUI development)

In this part , we will go through the essential steps regarding the technical processing and the details of the implmentation of our system within the MDP framework.We will be also analysing the system capacity of dealing with uncertanty and environlent changement.

Implementation

Several steps must be made to create a student advising system based on MDP in C++. First, the relevant states that describe the student's academic setting must be determined, followed by the creation of data structures or classes to capture these states. The student's activities inside the advising system must then be recognized, and data structures or classes must be developed to reflect these actions. To quantify the attractiveness of each state-activity combination, a reward function must be developed, and transition probabilities must be estimated depending on the action selected. To learn and calculate the ideal value function for the MDP, the Value Iteration algorithm must be constructed, and the Policy Extraction technique must be utilized to extract the optimal policy for decision-making.

An interface must be established for the student to engage with the advising system, and methods to manage user comments and update the MDP model must be included. Following these methods, a successful MDP-based student advising system may be constructed in C++.

We also implement the concept of OOP programming in our research implementation by separating our system into four main classes where each one representing a specific element of MDP design. This concept helps to assure a good representation of data and controls the access to the source code that defines the architecture of the system.

MDP Framework

The MDP framework is a mathematical tool that is used to simulate decision-making difficulties that arise in a random and sequential environment. It gives a systematic way to represent complex difficulties, allowing decision-makers to make educated decisions based on the probability of various outcomes. MDP is essentially a framework that can aid in the resolution of complicated decision-making problems by combining probabilistic transition probabilities and taking into account long-term repercussions.

An MDP's goal is to determine the best policy that maximizes the projected cumulative reward over time. A policy () is a state-to-action mapping that specifies the action to be done in each state. We also used the value iteration algorithm within this framework to solve the problem of transition and adapting to uncertainty.

Testing

In order to evaluate the system's performance in terms of accuracy, complexity and computational speed, we used the data presented before in this review to test our system. Evaluating a system's performance is critical to ensuring its efficacy and efficiency. This entails assessing how effectively the system operates in both simulated and real-world circumstances, determining if it satisfies the stated objectives, and finding any possible flaws or areas for improvement. A thorough assessment allows decision-makers to obtain insights into the system's strengths and shortcomings and make educated decisions to maximize its performance. This method can help to improve the system's overall quality.

The testing step is critical in guaranteeing the dependability, efficacy, and user satisfaction of an MDP-based student advising system. It aids in validating the system's implementation, evaluating its performance under various settings, and identifying areas for improvement in order to enhance decision-making and student assistance. Decision-makers may guarantee that the system achieves the required objectives, performs properly, and delivers a favourable user experience by undertaking extensive testing. This iterative testing and improvement method may assist drive continual system improvements, resulting in a more effective and efficient student advising system.

Results.

The testing phase and the results of our system based on the MDP framework can provide valuable insights into the system's accuracy, adaptability, handling of uncertainty, user satisfaction, performance metrics, and potential issues or limitations. These findings can guide iterative improvements to refine the MDP framework, algorithms, policies, or user interfaces to enhance the system's performance, accuracy, and adaptability. By gathering user feedback and assessing the system's performance under various conditions, decision-makers can continuously improve the system's capabilities and ensure it provides effective decision-making and student support.

While there isn't a specific student advising system based on the MDP framework that stands out as the most advanced as of September 2021, advancements in the field have been made using various techniques, including MDPs and reinforcement learning. These advancements include personalized course recommendations, adaptive decision-making, integration of real-time data, consideration of multiple objectives, predictive analytics and early warning systems, natural language processing and chatbots, and continuous improvement and learning. By leveraging these techniques, advanced student advising systems can provide tailored recommendations, adapt to changing contexts, identify at-risk students, and continuously improve their performance

Compared to other results of other student advising systems, we may notice a difference regarding the precision and the accuracy. This is essentially due to the impact of the data on the system architecture and the choosing of the variables that may represent the state and the action space.

MDP vs the conventional decision tree-based system

MDP (Markov Decision Process) and traditional decision tree-based systems are two methods to decision-making in several disciplines. MDP models express uncertainty openly through transition probabilities, making them appropriate for stochastic settings and probabilistic outcomes. MDPs are intended for sequential decision-making issues in which the optimum action at each stage is determined by taking into account the current state, available actions, and transition probabilities. MDPs, in conjunction with reinforcement learning algorithms, may be used to learn optimum policies through exploration and exploitation, adapting and improving over time. MDP models, on the other hand, can be complicated and difficult to comprehend.

Decision tree-based systems, on the other hand, do not naturally incorporate uncertainty and are often utilized for static decision-making situations. Decision trees are often simpler and more computationally efficient, although they may struggle with large-scale decision issues. Decision trees have a clear and understandable structure, which makes them ideal for classification and regression problems, consumer profiling, and rule-based expert systems.

The choice between MDP and a conventional decision tree-based system ultimately depends on the specific problem domain, the nature of the decision-making process, the availability of data, and the desired level of interpretability and adaptability.MDP is usually used to adapt to uncertainty and environmental changement which makes them suitable for long-term planning.

Conclusion

Finally, MDP and traditional decision tree-based systems are two distinct approaches to decision-making in a variety of disciplines. MDPs are intended for sequential decision-making issues and can deal with uncertainty via transition probabilities, whereas decision tree-based systems are better suited for static decision-making problems with clear rules and criteria. MDPs may be used in conjunction with reinforcement learning algorithms to learn and adjust optimum policies over time. This research helps to show the good interest in the MDP framework in the student advising system and explore the power of this approach in uncertainty and time adaption context.

References

- Augmenting Markov Decision Processes with Advising Loïs Vanhée, Laurent Jeanpierre, Abdel-Illah Mouaddib .
- An Illustration of the Use of Markov Decision Processes to Represent Student Growth (Learning) November 2007 RR-07-40 Research Report Russell G. Almond
- Conference: Explanation-aware Computing, Papers from the 2008 ECAI Workshop, Patras, Greece, July 21-22, 2008. The University of Patras.
- Generating Explanations Based on Markov Decision Processes. Conference: MICAI 2009: Advances in Artificial Intelligence, 8th Mexican International Conference on Artificial Intelligence, Guanajuato, México, November 9-13, 2009. Proceedings
- Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents International Journal of Computer Applications 181(1)