Faculdade de Ciências e Tecnologia
Universidade de Coimbra - DEI

Artificial Intelligence

Bayes networks in the development of agents to aid e-learning

Authors:

Inês Lopes Petronilho,2012137900,ilp@student.dei.uc.pt
João Pedro Santos Batanete, 2009113460, joaosb@student.dei.uc.pt

Introduction

Machine learning is a subfield of computer science that studies the ability of computers and machines to “learn” and adapt their courses of action according to the information gathered. The array of techniques used to implement it, and its applications, are countless at this point, but over the course of this project we will be focusing on an agent-driven approach with the incorporation of bayesian networks. These can be defined as probabilistic graphical models where the vertexes are random variables and the edges represent their conditional influence on one another, forming a directed acyclic graph(DAG). An example would be the modelling of a weather prediction system for a given day. By considering the cloudiness of said day, as well as other factors such as air humidity, as well as the estimated influence of these factors on the likeliness of raining or being sunny, we could create a bayesian network to support the decision-making agent in making its judgement.

One area of application that may benefit from the use of this technique is e-learning. By using a bayesian network to track a student’s performance in a given subject(or a group of student’s average performance), it can allow a teacher to make adjustments to the course’s structure as necessary, or provide students with extra feedback on those particular subjects.

Although it is hard to come across our exact theme, we have noticed that [2] mostly focuses on data exploration about student performance indicators, while ours will focus more on making the agent come up with solutions to teaching issues in real time. We will accomplish this by implementing the agent so that it provides the teacher with feedback relating to the data collected from the datasets provided with the student’s performance.

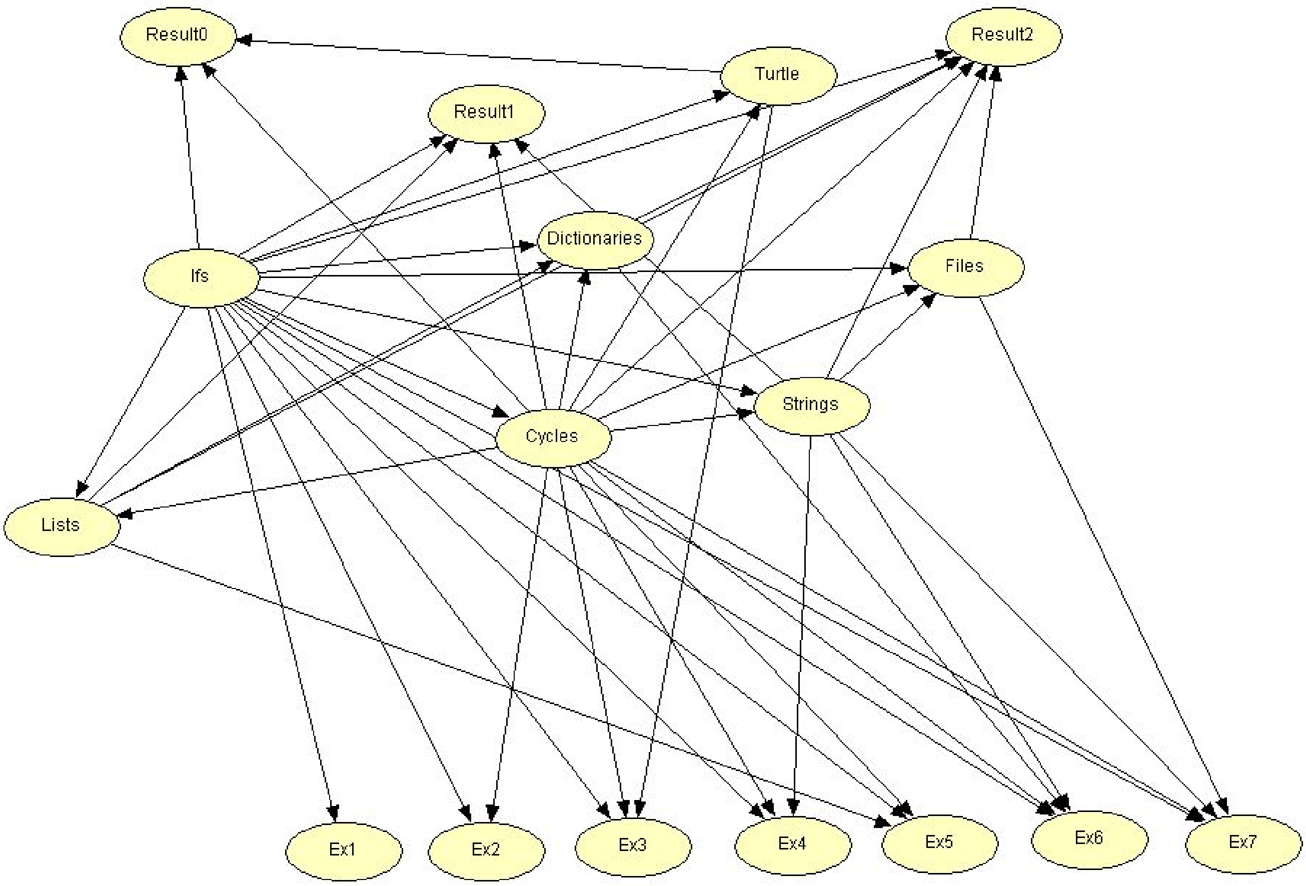
One advantage of the system lies in the fact that the system may at times be insecure about the answer and ask for the user’s input to solve problems may be seen somehow as a means of supervised learning, since the network will learn with time the appropriate answers based on the teacher’s own input. Another advantage of our approach could be the implementation of the selective attention model will be implemented. The goal of this model is to focus the agent’s senses on important information while simultaneously ignoring less relevant information, hence assuring that the recommendations are relevant.

Relevant information could be whether or not the student has completed a given exercise and how much time it took him to complete it, among other factors. By analysing the dataset provided, we could make an estimation on the likeliness that a given fact contributes to the student learning the subject successfully.

Even though the specific theme is quite hard to come across, there is a large number of implementations of bayesian networks for many programming languages, such as GeNIe for C++, PyCM for Python and Hugin(Java based). We will be using Hugin in the context of this project.

The remainder of the paper is structured as follows: the Overview where we will talk in more detail about the experience and its modelling, Results of the Experiment and further discussion of the results of the project, Conclusion and References.

Methods



Implementation

The agents shall be defined by a Bayesian network as defined by the diagram below. When it is determined that a student failed an exercise relating to a subject that he was supposed to know by then(example: if statements by mid term) an alert shall be provided to the teacher. The probability table is available later on in the report.

The Bayesian net above was created with *Hugin*.

The presented nodes fall into three categories:

- the knowledge of the students regarding all programming themes which are evaluated in the course (Ifs, Cycles, Turtle, Strings, Lists, Files and Dictionaries),
- the grades of the students in the exercises concerning those themes done over the year (Ex1-Ex7)
- three results which represent the expected condition of the student and the expected probability of him failing or passing the course at the beginning, middle and end of the year.

The several results are influenced by different parameters over the year. For example, at the beginning the student is only obliged to know about Ifs, and Cycles, while the last result is already dependent on all subjects.

The grade of each exercise gives information about not only the subject it is trying to evaluate, but also on later subjects, i.e., Exercise 2 depends not only of Cycles but also *Ifs*,

because it is highly unfeasible that a student would dominate cycles without having knowledge about *Ifs*.
We needed to estimate the initial knowledge that the student bears at the beginning of the year.

In terms of the technologies used, the selected language was Java, due to the group members having the most experience in it compared to the alternatives. The Hugin Java API was used to interact with the bayesian networks created.

The application can be run in test mode or normal mode. Test mode will print the confidence intervals, while normal mode generates alerts if the student is not apt to pass.
The tests were made by creating all possible combination of results and were posteriorly analysed to check if they made sense.

Results

The obtained results are submitted together with this report in the file outputs.txt.
The structure of said results is as follows:

- Each line i represents the output for test case i on file inputs.txt
- Each row j represents the confidence we have that the student is proficient in chapter j of the course. Chapters are ordered in the following order: Ifs, Cycles, Turtle, Strings, Lists, Dictionaries, Files. The last row represents the confidence that the student is proficient enough, given the phase of the semester we are currently at(beginning, mid-term or ending).

Discussion

In this section we will look to discuss the results obtained, and whether they validate the model used for the Bayesian net.

Looking at the results, one can easily notice that, as expected, a student's likelihood of being proficient at a given subject increases when he answers correctly to the exercises given to him. Obviously this is not enough to properly validate the model so let's take a closer look at some the results.

In the first test case(table 1), we assume the student manages to get all the exercises correctly and that we are at the beginning of the semester. As expected, the obtained confidence value is very close to 100%.

time	ex 1	ex 2	ex 3	ex 4	ex 5	ex 6	ex 7
beginning	correct	correct	correct	correct	correct	correct	correct

Table 1 - test case 1

On the other hand, we get the opposite result for test case in line 384(table 2), which assumes we are at the end of the semester and the student didn't get a single exercise right.

time	ex 1	ex 2	ex 3	ex 4	ex 5	ex 6	ex 7
end	incorrect	incorrect	incorrect	incorrect	incorrect	incorrect	incorrect

Table 2 - test case 384

Obviously, the two previous scenarios are very extreme, and the model would not even have to be close to be well implemented to answer them correctly. However, let's look at the test case in line 97(table 3). This test case assumes we are at the beginning of the semester and that the student somehow managed to get the Ifs and Cycles exercises wrong(the initial part of the course, which is necessary to understand the rest of the chapters). He did however get every other exercise right. The most intuitive explanation for this odd event is that the student is actually proficient in Ifs and Cycles but got the exercises wrong due to lack of attention of because the exercises themselves were trickier than usual. However, since he got everything else right, the model correctly assumes that he must still be proficient in the course(as well as Ifs and Cycles), albeit with a lower confidence value than if he got those two exercises correctly.

time	ex 1	ex 2	ex 3	ex 4	ex 5	ex 6	ex 7
end	incorrect	incorrect	correct	correct	correct	correct	correct

Table 3 - test case 97

This ability to respond to odd tests cases in a feasible manner significantly adds to the model's validity.

Generally, we can assume that our results are valid and make sense after further analysis. However, we are aware that the grading of an exercise may not be accurate. In a real case scenario, the grading of an exercise isn't binary (*true* or *false*). The grading falls somewhere between 0% and 100%.

Conclusion

Over the course of this project we have learned that Bayesian nets are powerful tools when it comes to predicting the behaviour of various complex systems, due to their versatility and power.

In this particular use case, and given the results we obtained during the test phase, we conclude that they can easily cover most of the requirements. Even though this project's main goal was to perform a proof of concept, one could easily apply the knowledge obtained here to solve the problem in a real life scenario using a maybe more complex Bayesian net(example: taking into account extra factor such as the number of hours of study, considering that the exercises have a percentual rather than a boolean value, etc).
Also, the parameters we used to make the tests were such that the probability tables associated with each node are quite subjective and were chosen according to our judgement, which may be different from the ideals of an experienced teacher, or the actual correct values.

However, one setback we encountered to the use of this kind of nets is the difficulty in coming up with correct values for the probability tables. Although in some cases (such as this one) this could be deduced from statistical data, it isn't always as simple.

References

1-https://en.wikipedia.org/wiki/Bayesian_network
2-http://www.sciencedirect.com/science/article/pii/S0888613X08000911 Russell G. Almond , Valerie J. Shute , Jody S. Underwood, Juan-Diego Zapata-Rivera: Bayesian networks: A teacher's view(2009)
3- http://iccm-conference.org/2012/proceedings/papers/0035/paper0035.pdf
4-The course's materials