GSoC 18 Project Proposal
Organisation : Aimacode
Aima-javascript : Graphical demos in aima-javascript

Basic Information

Name and Contact Information:

- **Name :** Surya Saini
- **Email :** sainisurya.1@gmail.com
- **Gitter Handle :** Dragneel7
- **Github :** Dragneel7

Education:

- **University :** Indian Institute of Technology, Roorkee
- **Field of Study :** Electronics and Communication Engineering (Batch of 2020)

Meeting:

- Available any time to discuss on gitter.Reachable easily through a mail.
- Wednesday, Friday available anytime between 2:00 pm IST(8:30 am GMT) to 12:00 am IST(6:30 pm GMT) and 6:00 pm IST(1:30 pm GMT) to 12:00 am IST(6:30 pm GMT) anyday.

Technical Skills:

I am very comfortable and proficient with the skills required for the project.
For instance, Javascript, I have an experience with libraries like D3.js, two.js and three.js. Other skills include Html/Css, Adobe Illustrator, Adobe Photoshop.
I have a strong background in web development, with proficient knowledge in version control system (like git), MTV and MVC architectures. These may not hold much importance to this project but helps me to establish a strong relation and understanding of the code I write.

About Me

I am 19 year old sophomore, currently enrolled in Electronics and Communication engineering at IIT Roorkee. I developed a passion for programming, web development and machine learning in my freshman year and from then, most of my time goes into reading new algorithms and papers, and implementing them. I have been reading about algorithms of AI for almost a year and have majorly focused on understanding the concepts elaborated by AIMA. My motive for participating in GSoC with Aima is to implement the concepts that I have learned, produce quality designs and visualizations to better understand the building blocks of Artificial Intelligence.

I have an experience of working closely with a team as I am an active member of **Information Management Group** at IIT Roorkee, a bunch of passionate enthusiasts who manage the **institute main website**, internet and intranet activities of the institute.

Projects

Rtograph :

Build with an aim to help the students search for their various needs in the campus, with a single search query and displaying the concerned place on a map, Rtograph can be aptly called the graphical wiki of IIT Roorkee.

Save The Clash :

Build as a part of the online round for Microsoft's Code.Fun.Do 2018, the app aims to help the visually impaired by providing them with a digital vision, using state of the art Image segmentation and depth estimation Networks.

Pre-GSoC Involvements

I have been working regularly on opensource for the past five months.
Following is a list of issues/PRs I've worked on so far with Aimacode :

Aima-javascript:

#133(**mergeable**)	Add performance measure for simple logical agents
#138(**merged**)	Fix issue link for A* search
#136(**mergeable**)	Add Min-Conflict Visualization
#153(**mergeable**)	Add visualization for Expectiminimax search

Aima-python :

#665(**merged**)	Fixed typographical errors in agents.py and agents.ipynb
#669(**merged**)	Updated README
#685(**merged**)	Added heuristics for A* function
#689(**merged**)	Solved issue of loading search.ipynb
#697(**mergeable**)	Added example for hill climb search

Aima-exercises:

#6(**merged**)	Added file lookup-table in README

Aima-glossary:

#4(**merged**)	Added glossary terms

Aima-data:

#13(**mergeable**)	Added quadragram file for hill climb search

HotOsm/Field-Campaigner:

#416(**merged**)	Fixed typographical errors in README
#419(**mergeable**)	Improved UX for campaign view
#420(**mergeable**)	Add real time user location on map
#417**(issue)**	Deploy the application in development mode.

Fossasia:

#310(**merged**)	Added exception handling for “Ask.com” web-scrape

Blogs

I started coding to visualize algorithms around mid-December 2017. These are the list of blogs that I have written so far. I try my best to be to create interactive visualizations, engaging them in this fun learning experience in gamified manner.

- Building blocks of decision - Logic
- Expectiminimax Search
- The 8 Queens Problem
- Understanding Rationality of Agents

When trying to design an algorithm, my very first step is to create a narrative to engage the reader. I search different sources to gain inspiration. My aim while designing is to be simple but elaborate at the same time. Besides technical aspects, I also write non-technical blogs, like this:

- God - A myth or reality

Project

Graphical demos in aima-javascript

My motivation for GSoC this year is to get started with the open source community. GSoC is a great program for introducing organisations with prospective contributors. The main reason that I wish to work with Aimacode is the vast opportunities to learn new algorithms of AI. Working with renowned names in this field will provide an environment to learn and grow. I began interacting with the community on gitter and have been regularly helping others and asking for suggestions whenever I get stuck. These past few months have been very thrilling, having a positive competition to motivate me. I strongly believe that with my skills and mentorship, I'll be able to contribute well to the organisation.

Deliverables:

I plan to implement visualizations for the following chapters:

- **Chapter 2 Intelligent Agents :** Introducing the reader to the concept of rationality using the vacuum world example. I aim to visualize the **remaining algorithm** for the chapter.

- **Chapter 6 Logical Agents :** This chapter tries to teach the reader the building blocks of decision making. It would be really interesting to implement logical agents as a tutorial based narrative. I plan to implement the wumpus world, engaging the user in tutorial based learning experience.

- **Chapter 14 - Probabilistic Reasoning:** I have a strong foundation for probabilistic applications in communication as a part of a my course curriculum . I plan to create interactive visualizations where user can play around with the parameters and have a better understanding, keeping in mind to incorporate real world experiences.

- **Chapter 18 - Learning from Examples:** This part of AI fascinates me a lot, teaching a machine from the pre-collected data. I have taken a few courses that explain various forms of supervised learning both graphically and mathematically. I also implemented an app predicting the scores of students as a part of OpenAI hackathon. My target for this chapter would be to design

visualizations where the reader can manipulate data and then show them results graphically, helping them to understand the changes in the system when the parameters are changed.
(The dataset to be used for the implementations will be decided later)
To implement the following I will be using NUMJS, Scikit Learn and other similar library.

Additional:
If time permits and the visualization(after discussion with mentor) are properly implemented, I will try to add the following:

- ❖ Add comments for the implemented chapters so that readers can ask and answer questions.
- ❖ Add definitions in aima-glossary for the implemented chapters.

I believe I have enough fuel to get started on my goals - as a result to my involvement with the organisation for the past 5 months. So I'll be setting grounds early to avoid stopgaps during the actual GSoC period. There are a couple of weeks before the actual timeline, I would make sure I have all my homework done by then.

Proposed Timeline
I believe I have enough fuel to get started on my goals - as a result to my involvement with the organisation for the past 5 months. So I'll be setting grounds early to avoid stopgaps during the actual GSoC period. There are a couple of weeks before the actual timeline, I would make sure I have all my homework done by then.

3 April - 9 April (Homework Week):
Give a brief reading to the chapters.
❏ Note down important points and examples that can be easily visualized.
❏ Think of some real world examples

10 April - 6 May(Pre Community Bonding Period):
❏ I'll be preparing for my end-term examination.
❏ I'll read the chapters and frame my design ideas into a doc.

7 May - 14 May(Community bonding period):
❏ discuss the project with the mentor.
❏ Submit a rough idea I prepared during the homework week and take reviews on that

15 May - 21 May(Week 1):
❏ Submit design doc for Performance Measures of Intelligent Agents
❏ Once reviewed and finalized, code the visualization

❏ Get the visualization reviewed
❏ Submit design doc for Logical Agents

22 May - 28 May(Week 2):
❏ Analyze the review of the design doc and incorporate the suggested improvements
❏ Implement the visualization and get the PR reviewed
❏ Submit the design doc for Conditional probability
❏ Implement the visualization

29 May - 4 June(Week 3) :
❏ Implement bayes Theorem

5 June - 11 June(Week 4) :
❏ Submit design doc for the revisited Wumpus World.
❏ Implement Wumpus world with probabilistic approach.

12 June - 18 June(Week 5 Community bonding 2) :
❏ Cover up the backlog and finish the unimplemented designs(if any)
❏ Get review from the mentor about phase one and discuss the algorithms to be implemented in phase 2 and the approach to their implementation

Phase 1 Milestone reached
19 June - 25 June(Week 6) :
❏ Submit a doc of Decision Trees for review
❏ Implement the design doc and add a narrative why learning came into play, incorporating sections 18.1 and 18.2, i.e., Forms of Learning and Supervised Learning.

26 June - 2 July(Week 7) :
❏ Find a suitable dataset for Linear regression(Like Stock price estimation)
❏ Submit design doc and implement Linear Regression

3 July - 9 July(Week 8) :
❏ Implement Logistic regression (Using the same dataset)
❏ Submit a design doc for Artificial Neural Networks

10 July - 16 July(Week 9) :
❏ Implement Artificial Neural Networks

17 July - 23 July(Week 10 Community Bonding 3) :
❏ Cover up the backlog.

❏ Get review from mentor about phase 2 and discuss the algorithms to be implemented in phase 3 and their approach to implementation

Phase 2 Milestone reached
24 July - 30 July(Week 11) :
❏ Submit design doc for KNN
❏ Implement KNN
❏ Submit design doc for Support Vectors Machine

31 July - 6 August(Week 12) :
❏ Implement Support Vector Machine and graphically depict how they exactly work
❏ Submit a design doc for Practical Implementations

7 August - 14 August(Week 13) :
❏ Implement Practical implementations with MNIST digit recognition

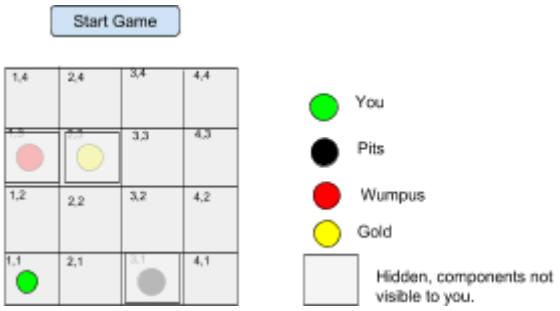
15 August - 21 August(Conclusion) :
❏ Buffer week and conclusion of GSoC 2018.

Availability
My vacations are from 5 May to 15 July. During this time period I'll be able devote 40-50 hours a week till my college reopens and 30-35 hours per week after that. I am free on weekends.
My approach would be to create design docs separately for each algorithm and code the once approved design docs. Considering with prior understanding and experience with the above stated chapters I believe I'll be able to complete these in the alloted time period.

Building Blocks of Decision:
How does a machine navigate to a place in an unknown environment?
To understand how a machine would find its way, let us understand how humans do the same thing by working our way through one imaginary world, the wumpus world. As we progress, we will understand the concepts that help in this decision making.

The Wumpus World (Prototype)
Imagine yourself to be in world full of caves, one cave connected to another via doorways. One of them having tons of gold. Clearly, you want it for yourself. What do you do?

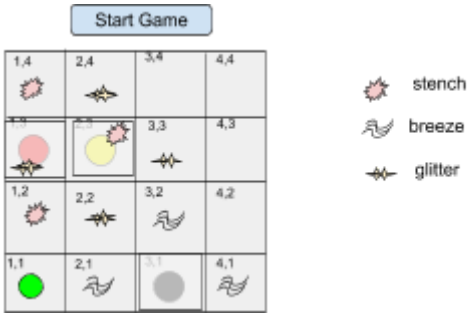
Simply navigate around the caves to find the gold. Sounds easy isn't it!!. But there is a catch, not all the caves are safe, some contain a pit(and of course you wouldn't want to fall in a pit) and one of these caves is home to a terrible beast named **Wumpus**, which will kill you if it encounters you on your adventure.



Just navigate around and maybe if you are **lucky** enough you'll find the gold.
So how was the experience. Not so good probably.
But when machines are making important decisions, nothing can be left out to chance. We need a concrete decision making algorithm, one that factors in outside information to reach the goal.

The good thing is like the real world, the wumpus world environment isn't solely based on luck. You have the following information about the world to help you.

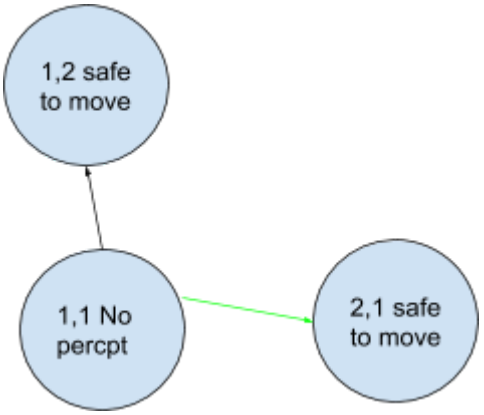
1.) In the cave containing the wumpus and in the directly (not diagonally) adjacent caves, the agent will perceive a **Stench**.
2.) In the cave directly adjacent to a pit, the agent will perceive a **Breeze**.
3.) In the cave where the gold is, the agent will perceive a **Glitter**.
4.) A cave can have only one of the above.



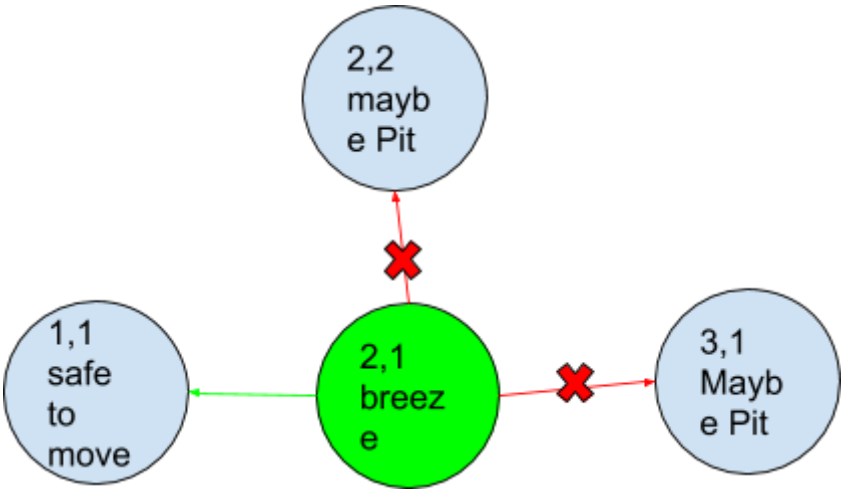
Feeling confident now? So let's navigate around a second time and this time **logically** plan our next moves.
Congratulations if you found the gold. The information often called **percepts** you had about the environment might have helped you navigate through the wumpus world with a confidence on your choice to navigate to.

When you started the game, you were at cave (1,1) what percept did you get?

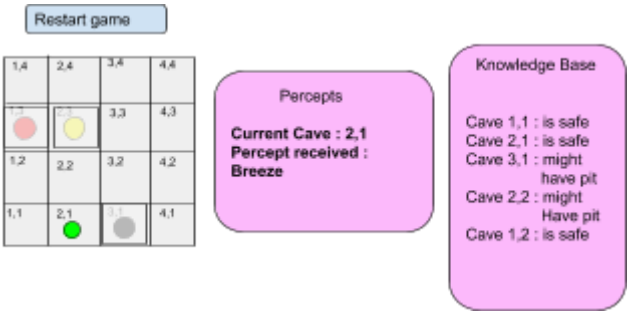
None ! Logically you concluded that cave (1,2) and cave (2,1) are safe to navigate.



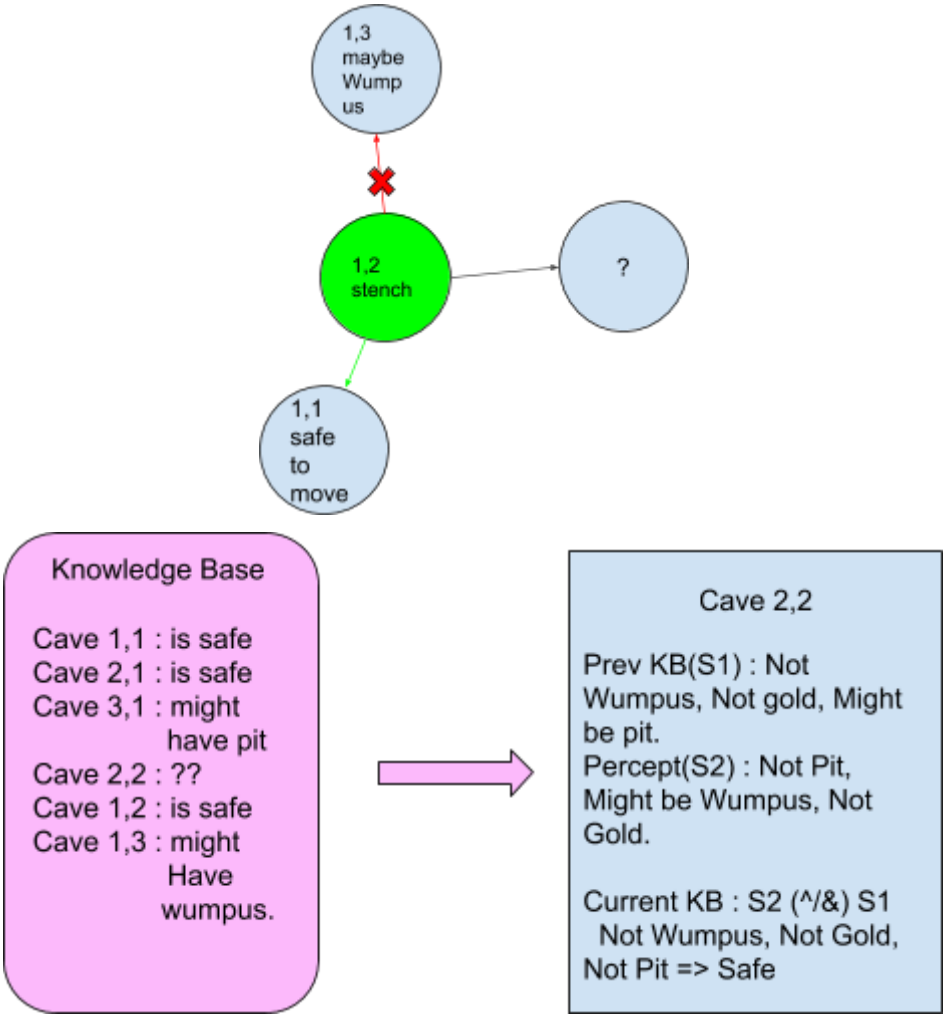
You now have two safe positions to navigate. Let's say you moved to cave (2,1). Cave (3,1) has a pit, so you must have received a **Breeze.**
So what do you conclude from this percept?



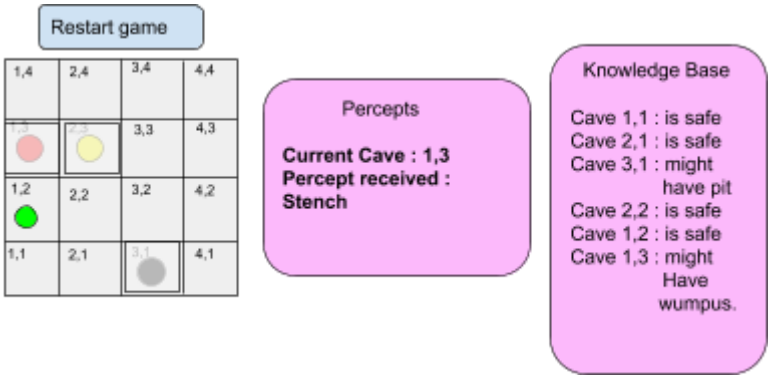
There might be a pit in any of the 3 adjacent caves. You surely **remember** that cave (1,1) was safe.
The memory or knowledge you have of the environment is called the Knowledge Base. This Knowledge base is updated at each step and helps you navigate through the caves.



Your next move brings you back in cave (1,1) from where you can navigate to cave(1,2). As the cave (1,3) has Wumpus in it. You perceive a **Stench.**

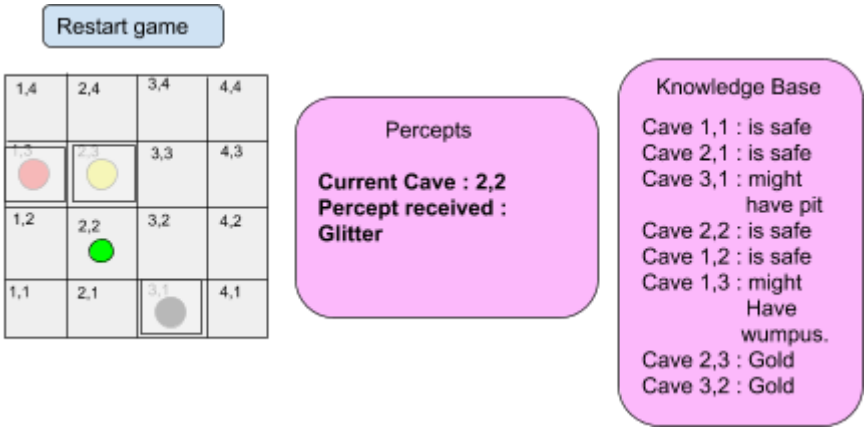


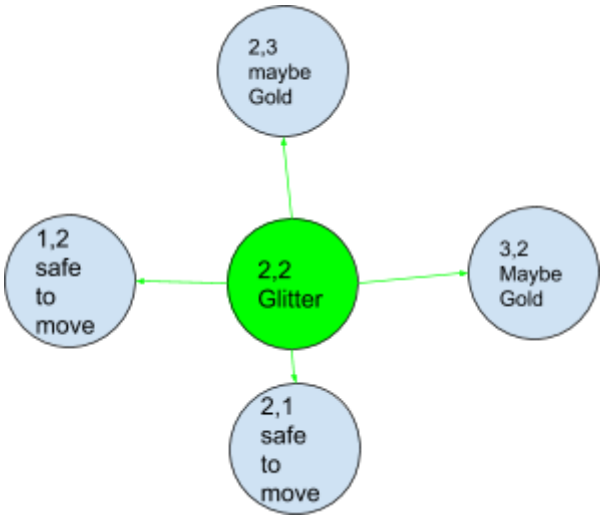
Cave (1,1) is safe. (Moving there will lead you into a loop)
Cave (1,3) might have a Wumpus. (Not good to navigate there)
Cave (2,2) ??
Our previous Knowledge base said that cave(2,2) might have a pit. Our current percept tells us that the cave(2,2) might have a wumpus.
Both of these statement cannot be true at the same time.
Therefore you can logically argue that cave(2,2) is **safe**.



NOTE: “Logical-and(&) works in wumpus world because a single cave can have only one possibility. In real world there are more than 1 possibilities, ‘Logical and’ will still work but we will require more information about that place. ”

By now you know where the pit was, you even know the position of the wumpus.
A single information helped you deduce all this information because you were logical in your approach. Now the best move is to navigate to cave(2,2), where you receive **Glitter.**

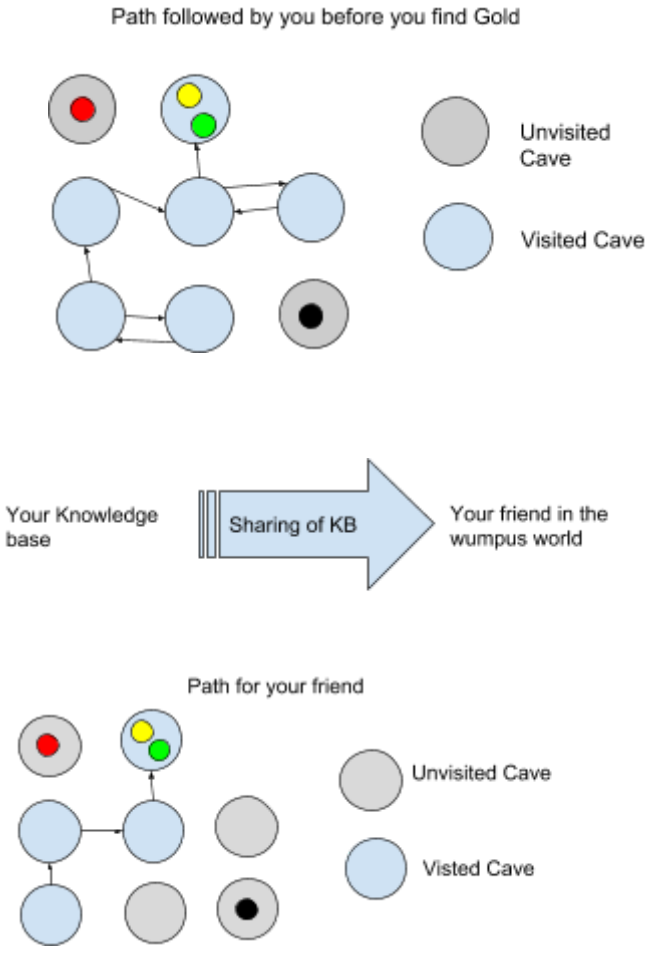




Navigating to cave(2,3) takes you tons of gold and you Win. Navigate to cave(3,2) and you will still win but will take a few moves extra.

Thus getting percepts and logically operating on them helped you navigate the caves and find gold.
This same logic can be applied to machines so that they too can logically navigate the wumpus world with efficiency equal to or greater than humans.

Know suppose your friend also wants to navigate the caves and get the gold. So would you want him to navigate the caves fearing pits and the wumpus? No, you wouldn't. You can help him by sharing your knowledge base with him.
Knowledge once acquired can be passed on to others which can then navigate the environment with higher efficiency.



Conclusion:

Navigation through the wumpus helps us build basic concepts of logic required for navigation in an unknown environment. In the real world the percepts received would be different and varied, and consequently the logic applied to analyze them would be more complex. But even the complex logic is build upon the simpler, more basic logical foundation that we have discussed.