

Lab 8 – Genetic Algorithms & AI

1. **Study the following paper that uses GA to solve the Knapsack problem.**

http://www.micsymposium.org/mics_2004/Hristake.pdf

Make sure you understand it very well. The main goal here is to enable you to understand some published paper well and relate it to the content of the lecture covered in the class. This paper will also enable you to do some research work as would be needed in future e.g. when you join an industry.

Nothing is due for this, BUT your understanding will be helpful to answer questions on GA in the Final exam. Also, your understanding of the paper will help you solve Problem #2 below.

2. **You have 5000 MP3 files sitting on your computer's hard disk. Unfortunately, the hard disk has started making noises, and you decide that you had better back up the MP3s. Equally unfortunately, you can only burn CDs, not DVDs, on your computer. You need to minimize the number of CDs that you use, so you decide to design a genetic algorithm to choose which MP3s to put onto each CD in order to fill each CD as completely as possible. Design a genetic algorithm to solve the problem. You will need to consider how you would encode the inputs, which genetic operators are suitable, and how you would make the genetic algorithm deal with the fact that you have multiple CDs, not just one CD. (Note: You can use some ideas from #1 above).**

Ans: Knowing that each CD, standard 120mm, has a capacity of 700MB is obvious that we need several CDs to do the backup (each mp3 song usually is approximately 5MB, depends of the length in minutes and the coding format used).

Considering that we don't have enough information about the size of the songs, the average or the benefit for the owner (how much he/she likes each song in this case), but it turns irrelevant because all songs wants to be backed up, we will have to do some considerations:

- We can use the solution exposed in the paper about the Knapsack problem, because we need to maximize the songs being backup up into each CD. In this case the Knapsack capacity is similar to the CD capacity.
- The only difference is that we have several CDs (like several Knapsack), but we can take one CD at the time (700 MB of capacity) and solve the problem of how many files can we storage in it, trying to maximize the amount (considering all songs with the same benefit in this case).
- Later take the other CD and the remaining songs and repeat the cycle, we can to consider a maximum quantity of CDs denotated as K.
- Population size: Considering the initial number of files is 5.000, it would be too much to be considered as the population, and as we mentioned before with a standard encoder each song could be 5MB we can to reduce the estimated population to $700/5=140$ songs, so we can try with an initial population of 200 and try to maximize the quantity of songs in the CD. 5.000 would be too much and would make slower our GA solution.
- We would define our chromosomes as in the paper, representing each in an array of Benefits (B) and Size (S) tuples:

0	1	2	3
B1,S1	B2,S2	B3,S3	B4,S4

And encoding to 0s and 1s, indicating if each file goes or not into the CD (let's call it CD i for each iteration):

0	1	2	3
1	0	0	1

In this example we would be starting with files 1 and 4 being included, and files 2 and 3 excluded from CD i .

- Later we ran the GA, using one of two methods (roulette-wheel for example or group selection with elitism option) to check how much files maximize the CD capacity.
- When we obtain the resulting number of files we must reduce the population in the same quantity and continue iteration with the next CD ($i+1$) and the remaining population of files.

At the end we would have N (or K CDs if we reach the limit) with a quantity of songs near to its maximum, possible not the most optimal solution but must be close to it and with low computational effort (linear and not exponential), solving so our NP problem of backing up our songs.

3. For each of the following Agents, develop a PEAS description of the task environment:

- Robot soccer player
- Internet book-shopping Agent
- Autonomous Mars rover
- Mathematician's Theorem Proving Assistant

Task	Performance measure	Environment	Actuators	Sensors
Robot soccer player	Obey the rules, maximize the score, numbers of mistakes	Grass, ball, team players, rivals	Head, body, arms, legs, speakers	Touch sensors, video camera, microphone, communication sensors
Internet book-shopping Agent	Time, minimizing cost, books that some customer are interested	Customer, Browser, internet	Display, speakers, add new orders	Mouse, microphone, web pages, buttons or hyperlinks, keyboard,
Autonomous Mars rover	Explore terrains, collect, analyze, explore, communication with Earth station	Vehicle, mars	Communication devices, arm, legs	Video camera, receivers, communication sensors
Mathematician's Theorem Proving Assistant	Correctness in proving, time requirement	The theorem to prove, mathematical tools	Display, speakers, output functions, output axioms	Mouse, Keyboard, microphone, functions / methods to take input from mathematical tools

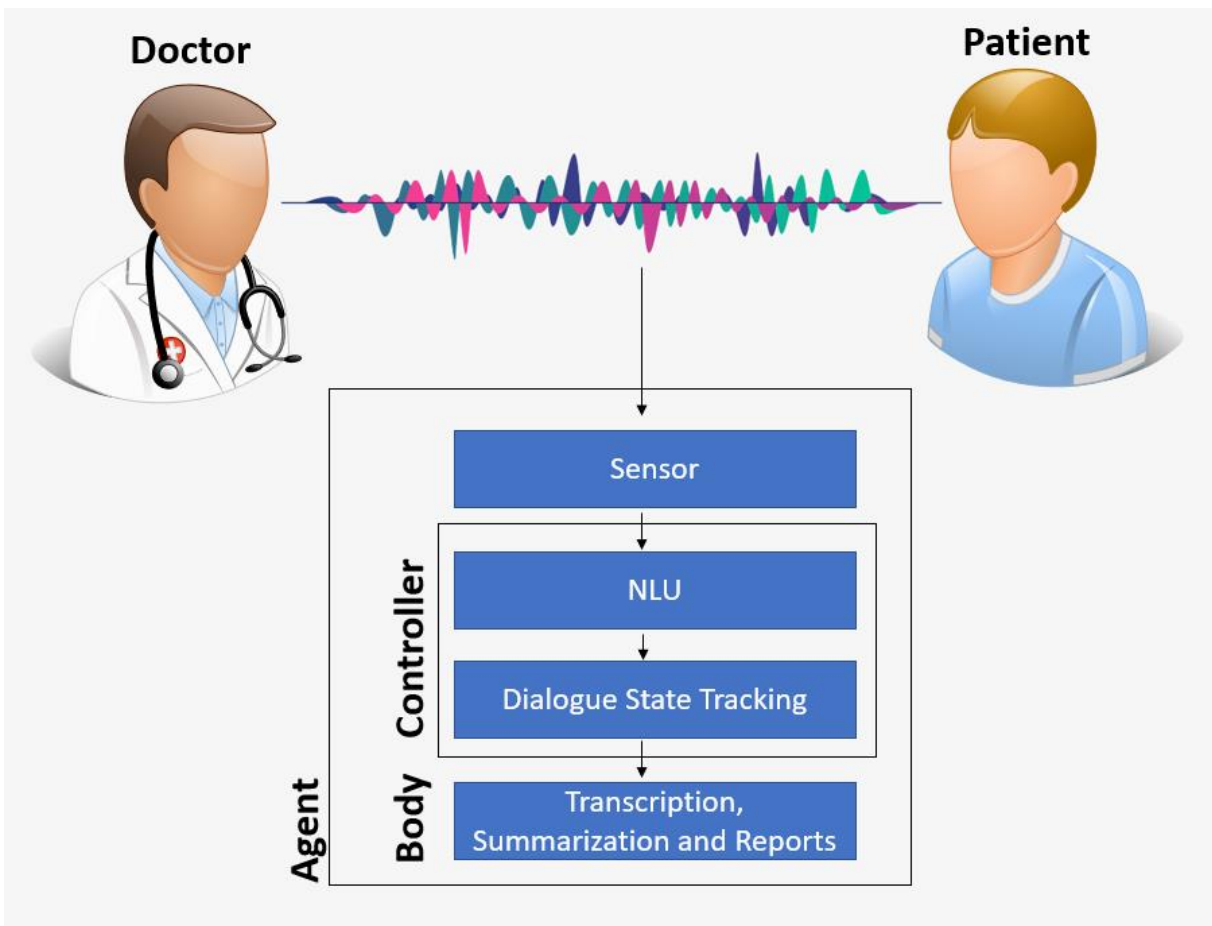
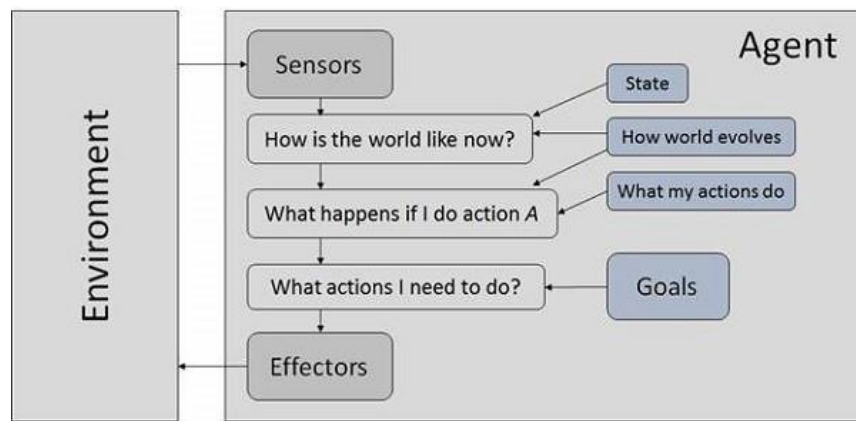
4. Consider a Natural Language based Medical Assistant Agent.

Draw its architecture along with the environment. Mention important goals and utility functions.

Ans: To design the architecture along with the environment of the Agent, as well as to define its important goals and functions, we start describing the agent goal:

The agent will understand conversations using NLU models with medical vocabulary. Previously trained on medical terminology and data, the agent would be able to listen conversations between a doctor and patient (consensually) to transcribe, summarize the conversation as notes (for future reference) and create structured reports. This would reduce the demand of healthcare personnel, allowing them to invest their time in other tasks that directly improve patient's experience and wellness.

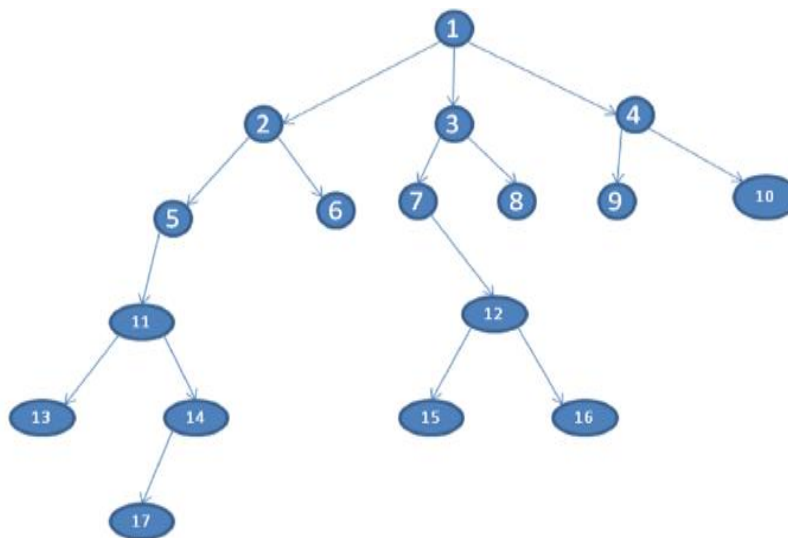
Thus, considering the statement above, the medical agent should be a Goal-Based Agent with the following architecture, environment and characteristics:



Agent Characteristics:

Description	Type	Goal	Some Issues
Medical Assistant Agent	Goal-Based	Transcribe, summarize, and create structured reports about doctor and patient conversation to reduce demand of healthcare personnel	<ul style="list-style-type: none">• Understand the semantics of conversation sentences• Using the facts to identify and register different things within the conversation: what were the symptoms presented, which was the treatment indicated, any medication prescribed...• Multi-level goal states: transcription completed, summary build, report created.

5. For the following tree, list the order in which the nodes are visited using Depth-First Search.



Ans: Using Depth-First Search, nodes will be visited in the following order:

1, 2, 5, 11, 13, 14, 17, 6, 3, 7, 12, 15, 16, 8, 4, 9 and 10.

6. The travelling salesman problem (TSP) is an NP-Complete (i.e. complex – needs exponential time to solve). TSP asks the following question: "Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city and returns to the origin city? Try to solve this problem using GA. Then implement your solution in Python. You would need to import numpy, pandas, random and operator.

Ans: (view file geneticalgorithm.ipynb)