**CSE-241: Artificial Intelligence**

**End-Semester Exam - Even Semester 2019-20**

**Total Marks: 60**

**Name:** Anwoy Chatterjee

**Roll No.:** 18075075 (B.Tech)

**Department:** CSE

**1 (a) From the point of view of Knowledge Representation, what kind of information does it contain, compared to, say, an essay or matrix or a family tree. What is the significance of your answer for the automation process? [4 Marks]**

**Answer:**

Knowledge Representation is a very vital part of an AI problem.

Here if we compare the knowledge representation for our CV with essays, we can say that essays are more of a generalized representation and stores a bulk information without proper categorization. With respect to the family trees, we can say that family trees are basically used to represent data in which there exists some kind of inter-dependence or correlation between different labels or generations; whereas CVs contain some chunks of information and no such prominent inter-dependence exists in it.

The best AI data-structure or form of knowledge representation which can represent CV is probably the ‘**Frames**’. Frames divide the information into many sub-structures or categories which helps us in representing our CV and divide the total content into various fields like the Contact Information, Education, Skills, Projects undertaken etc.

When we try to automate the process of say Resume screening, the Knowledge Representation plays a major role. The more relevant and structured is the representation, the easier it would be for the AI agent to extract the required features out of the CV. Apart from accelerating the process of feature extraction, a good knowledge representation will also help to increase the accuracy of the process by decreasing the chances of missing or mis-classification of any information or feature by the AI agent.

**1 (b) Represent your CV in HTML and XML. Which one is better for AI and why? [8 Marks]**

**Answer:**

**CV in XML**

<? xml version= "1.0" encoding= "utf-8" ?>

<? xml-stylesheet href= "CV.xslt" type= "text/xsl" ?>

<CV>

<Candidate

Name= "Anwoy Chatterjee"

Current Address= "Dr. C.V. Raman Hostel, Room no.-21, IIT(BHU), Varanasi"

Mobile= "9064612417"

Email= "anwoychatterjee.cse18@itbhu.ac.in"

Date of Birth= "24.10.2000">

I am a Sophomore pursuing B.Tech at <strong> Indian Institute of Technology (BHU) Varanasi </strong> in the Department of Computer Science and Engineering.

</ Candidate>

<Education>

<academic record

institute= "IIT (BHU), Varanasi"

from= "2018"

to= "Present"

exam= "Semester 1,2 & 3"

CPI/Percentage= "9.74" >

</academic record>

<academic record

institute= "Madhuban Goenka Vidyalaya"

from= "2016"

to= "2018"

exam= "Higher Secondary Examination"

CPI/Percentage= "97.50" >

</academic record>

<academic record

institute= "Vivekananda Siksha Niketan High School"

from= "2005"

to= "2016"

exam= "Secondary Examination"

CPI/Percentage= "95.28" >

</academic record>

</Education>

<Technical Skills

Languages= "C, C++, Python, HTML, CSS, SQL"

Frameworks= "Keras, TensorFlow, Django"

Topics of interest= "Machine Learning, Deep Learning, Computer Vision">

</ Technical Skills>

</CV>

**CV IN HTML**

< html >

< body >

< div >

< p >Anwoy Chatterjee< br > Dr. C.V. Raman Hostel, Room no.-21, IIT(BHU), Varanasi < br >9064612417< br > anwoychatterjee.cse18@itbhu.ac.in </ p >

</ div >

< div >

< h1 >CV</ i ></ h1 >

< div >

< h1 >< i >< span >ANWOY CHATTERJEE</ span ></ i ></ h1 >

</ div >

< div >

</ div >

< div >

< pre >< h1 >OBJECTIVE:</ h1 >

In anticipation of an opportunity providing a challenging work in the field of computers< br >to enrich my knowledge and enhance my skills.

< hr >

< h1 >EDUCATION:</ h1 >

< table >

< tr >< th > Institution </ th >

< th >Examination</ th >

< th >Board/University</ th >

< th >CPI/Percentage</ th >

</ tr >

< tr > < td >IIT(BHU), Varanasi</ td >

< td >Semester 1,2 & 3 </ td >

< td >None</ td >

< td >9.74</ td >

</ tr >

< tr >

< td > Madhuban Goenka Vidyalaya </ td >

< td > Higher Secondary Examination </ td >

< td >WBCHSE</ td >

< td >97.50%</ td >

</ tr >

< tr >

< td > Vivekananda Siksha Niketan High School </ td >

< td > Secondary Examination </ td >

< td >WBBSE</ td >

< td >95.28%</ td >

</ tr >

</ table >

< hr >

< h1 >TECHNICAL SKILLS:</ h1 >

Languages: C, C++, Python, HTML, CSS, SQL

Frameworks: Keras, TensorFlow, Django

Topics of interests: Machine Learning, Deep Learning, Computer Vision

</ div >

</ body >

</ html >

The XML format is preferred for AI problem. This is so because the tags or the namespace labels in XML represent the category of information which makes it easier for the AI agent to extract features. Also, XML is more flexible allowing us to use our own labels which makes our CV more structured and easier to be interpreted by the AI agent, compared to the corresponding HTML format.

**1 (c) Can you represent the same CV in Predicate Logic? If yes, try to do that (partly, just to show that it is possible or not). [8 Marks]**

**Answer:**

Yes, the useful fields or categories in CV can be represented using predicate logic.

Example:

Say the “Technical Skills” section of my CV, as shown in previous question, includes some frameworks, say, “Keras, TensorFlow, Django”.

To represent this using predicate logic, we can consider our ‘Universe of Discourse’ to be U={ ‘Keras’, ‘TensorFlow’, ‘Django’ }.

Now the “Technical Skills” section of my CV can be represented as:

∀x F(x) or, ∀x ϵ U, F(x)

where, F(x) represents the statement, “I am skilled in x”.

**2 (a) Formulate a problem definition for the AI agent that will extract information from the CV and process it for screening. [4 Marks]**

**Answer:**

Here we are asked to both extract information from a given CV and process it for screening. This involves the problems of both ‘Resume Parsing’ and ‘Resume Screening’.

As a part of Resume Parsing, our aim will be first to convert the raw unstructured resume into a structured format, say into XML format and thereby segregating the whole CV into some fields like the contact information, educational background, skills, projects undertaken etc. This facilitates the work of the AI agent which in turn aims to build feature vectors out of the structured CV available in XML.

Now for ‘Resume Screening’, the problem statement can be formulated as predicting a score based on the feature vectors extracted from the CVs of each candidate. It can be viewed as a Regression problem based on Supervised learning, where training of the model can be performed using numerous data collected from the CVs of the already interviewed candidates in the past, and the reference score is the score given to them by the recruitment committee consisting of human experts.

**2 (b) Can you apply some machine learning algorithm to solve the problem as formulated by you? If yes, outline the solution, mentioning the algorithm. No need to write the pseudocode. [10 Marks]**

**Answer:**

Yes, we can apply machine learning algorithm to solve the problem.

What the AI agent basically does is based on the requirements of the post for which recruitment is done, it can be trained on resumes of previously recruited candidates from which it extracts say 50 words which appear most frequently in each of the fields of skills, projects etc. of the CV. Once that words have been identified, it builds feature vectors out of the submitted CVs which basically consists of the frequencies of those identified words. We can add some more numerical and Boolean features to our feature matrix, like the number of years of work experience, or Boolean features like whether the candidate possess a certain required skill or not.

We can construct a hypothesis function which basically asserts a score to the CV given as input. Our model can be trained using the training set consisting of the similar feature vectors xi(k) , described above, for CVs of past candidates and the score given to them by the expert recruitment committee, which is our yi. As we train using lots of data, the weights gets adjusted according to the need of a particular skill (included in feature space as word appearing frequently in sample CVs) for the post and is thereby relevant in terms of recruitment perspectives.

Our problem is thus now a regression problem, and numerous supervised learning algorithms are available to solve it. We can use the Linear Regression, Support Vector Regression algorithms, or any other well-established algorithm for regression problems.

The candidates with the highest scores can be recruited based on the number of posts available for the profile.

**2 (c) How will you evaluation the performance of this AI agent for evaluating the CVs? [6 Marks]**

**Answer:**

Our AI agent is basically a supervised learning model. So, to measure the performance of our model, we can use any error function like the mean-squared error, or, R-squared error. We can assign any threshold error value based on our permissible error for the evaluation of relative performance of the model.

As an alternate approach, we can divide the initially available training data into two parts using train-test split approach, train the model on the larger section of data and evaluate its performance by calculating the mean error on the other section. We can also use k-fold cross validation for this purpose.

**3 (a) Recall the Time Table task that you were assigned for the lab component of this course. Can you use the A\* algorithm to solve that problem? If yes, give an outline of how you will do it. Is A\* completely adequate for solving this problem? If not, what are the problems and what could be the possible solutions? [15 Marks]**

**Answer:**

Though we can solve the problem using A\* in some cases, but it is not adequate for solving every instance of the problem.

A\* algorithm is an informed search algorithm in which we need to assign some cost to each node of the search tree which acts as the heuristic to determine which node is to expanded in the next step of the algorithm.

We can create a search space consisting of all possible time-tables and our aim is to search for a minimum cost path which leads from the root node in the search tree to a goal node corresponding to a time-table which satisfies all the constraints. So, we will start from the root node and expand it’s only those children which has not been expanded before. We then add the paths we get for each child to a stack, sort the stack in descending order of the cost of the last node of each element, pop the element at the top of the stack, and do the above said procedure for its children. We continue doing this until the popped element (which is basically a time-table) satisfies all the required constraints.

**Problems:**

However, construction for the search tree from the search space of all possible time-tables doesn’t seem to be practically feasible for a large number of subjects and time slots. Moreover, the construction of search tree should follow some pattern which makes searching optimal, and getting patterns in a pool of time-tables may not be practically possible.

Also, when we consider A\* we as we focus more on getting a minimum cost path i.e. searching the goal node as fast as possible, it may not guarantee that we will get the best possible time-table.

**Solution:**

Instead of using informed search algorithms like A\*, I think it would be more a practical and better approach to use Genetic Algorithms for solving this constraint-bounded problem of preparation of Time-tables.

**3 (b) Why do you think there is no ready-made tool available to solve this problem, whether using AI or not, given that it must be quite a common problem for a lot of people? At the same time, second year students are trying to solve problems such as designing vaccines, which is supposedly a much harder problem. Why do you think this strange situation exists? [5 Marks]**

**Answer:**

First of all, the knowledge representation for this time-table problem is not so simple as it involves a lot of constraints based on the number of students, number of subjects, time slots, number of professors or teaching assistants available, available number of classrooms etc. So, if we want to solve this using AI it becomes a much complex problem.

Secondly, it is a NP hard problem when we consider it to get solved by any non-AI algorithm. Consequently, it cannot be solved it optimum time, neither is the generation of a solution guaranteed.

Now, coming to the point of its comparison with designing of vaccines, this problem is a secondary problem and is not as crucial as designing of vaccines which, though is harder, is essential specially nowadays when our lives are at a stake. Automation of the time-table problem is not a crucial topic undertaken as research as it is often more beneficial to solve the problem manually, because a time-table, although if it satisfies all the required constraints, may need to be altered due to some considerable problems of the teacher or students.

----- o o o -----