Machine Learning lecture 5

Learning denotes changes in a system that enable a system to do the same task more efficiently the next time.
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.
Learning denotes changes in a system that enable the system to do the same task more efficiently next time.
Learning is an important feature of “Intelligence”.

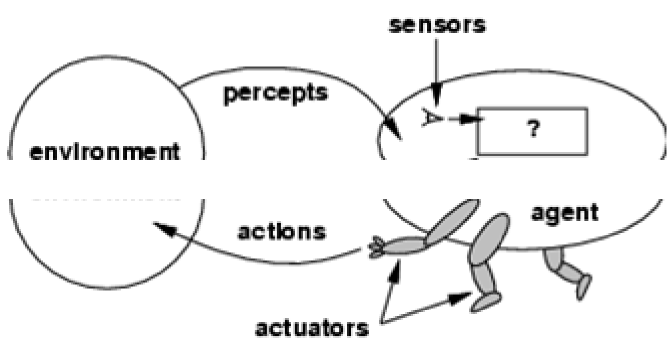
Definition
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.
This means : Given : A task T A performance measure P Some experience E with the task
Goal: Generalize the experience in a way that allows to improve your performance on the task.

Why do you require Machine Learning ?
■ Understand and improve efficiency of human learning.
■ Discover new things or structure that is unknown to humans.
■ Fill in skeletal or incomplete specifications about a domain.

Learning Agents
An agent is an entity that is capable of perceiving and do action. An agent can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.

Intelligent Agent (Learning Agent) Agent is an entity that is capable of perceiving and do action. In computer science an agent is a software agent. In artificial intelligence, the term used for agent is an intelligent agent.
Learning is an important feature of “Intelligence”
Percept : agent's perceptual inputs
Percept sequence : history of everything the agent has perceived
Agent function : describes agent’s behavior
Agent program : Implements agent's function

Learning Agent consist of four main components :
◊ Learning element,
◊ Performance element,
◊ Critic, and
◊ Problem generator.



Components of a Learning System
■ Performance Element: Is the agent itself that acts in the world. It takes in percepts and decides on external actions.
■ Learning Element: It responsible for making improvements, takes knowledge about performance element and some feedback, determines how to modify performance element.
■ Critic: Tells the Learning Element how agent is doing (success or failure) by comparing with a fixed standard of performance.
■ Problem Generator: Suggests problems or actions that will generate new examples or experiences that will aid in training the system further.
Example : Automated Taxi on city roads
■ Performance Element: Consists of knowledge and procedures for driving actions. e.g., turning, accelerating, braking are performance element on roads.
■ Learning Element: Formulates goals. e.g., learn rules for braking, accelerating, learn geography of the city.
■ Critic: Observes world and passes information to learning element. e.g. , quick right turn across three lanes of traffic, observe reaction of other drivers.
■ Problem Generator: Try south city road .

Paradigms of Machine Learning
■ **Rote Learning:** Learning by memorization; One-to-one mapping from inputs to stored representation; Association-based storage and retrieval.
Rote learning focuses on technique avoids memorizing the understanding material so that the inner complexities but it the can be recalled by learner exactly the way it was read or heard.
•Learning by Memorization which avoids understanding complexities the subject that is being learned; Rote learning instead focuses on memorizing the material so that it can be recalled by the learner exactly the way it was read or heard.
•Learning something by Repeating over and over again; saying the same thing and trying to remember how to say it; it does not help us to understand; it helps us to remember, like we learn a poem, or a song, or something like that by rote learning.
■ **Induction**: Learning from examples; A form of supervised learning, uses specific examples to reach general conclusions; Concepts are learned from sets of labeled instances.
■ **Clustering**: Discovering similar group; Unsupervised, Inductive learning in which natural classes are found for data instances, as well as ways of classifying them.
■ **Analogy**: Determine correspondence between two different representations that come from Inductive learning in which a system transfers knowledge from one database into another database of a different domain.
■ **Discovery**: Learning without the help from a teacher; Learning is both inductive and inductive theorems when deductive. It is deductive if it proves theorems and discovers concepts about those theorems. It is inductive when it raises conjectures (guess). It is unsupervised, specific goal not given.
■ **Genetic Algorithms**: Inspired by natural evolution; In the natural world, the organisms that are poorly suited for an environment die off, while those well-suited for it prosper. Genetic algorithms search the space of individuals for good candidates. The "goodness" of an individual is measured by some fitness function. Search takes place in parallel, with many individuals in each generation.
■ **Reinforcement**: at Learning from feedback (+ve or -ve reward) given end of a sequence of steps. Unlike supervised learning, the reinforcement agent cannot learning directly takes place compare the in an environment results of its where action to the a desired result. Instead, it is given some reward or punishment that relates to its actions. It may win or lose a game, or be told it has made a good move or

a poor one. The job of reinforcement learning is to find a successful function using these rewards.

Version Spaces
A Version Space is a hierarchical representation of knowledge that keeps learning track of all useful information supplied by a sequence of version space without remembering any of the examples
The version space method is a concept learning process.

Version Space Characteristics
Represents all the alternative credible descriptions of a algorithmic. A plausible description is one that is applicable to all known +ve examples and no known -ve example.
■ A version space description consists of two complementary trees:
- contains nodes connected to overly general models, and other,
- contains nodes connected to overly specific models.

■ Node values/attributes are discrete.

Assumptions
■ The data is correct ie no erroneous instances.
■ A correct description is a conjunction of some attributes with values.

Decision Trees
Decision trees are powerful tools for classification and prediction. Decision trees represent rules. Rules are easily expressed so that humans can understand them or even directly use in a database access language like SQL so that records falling into a particular category may be retrieved.

Description
■ Decision tree is a classifier in the form of a tree structure where each node is either a leaf or decision node.
- leaf node – indicates the target attribute (class) values of examples.
- decision node – specify test to be carried on an attribute-value.

■ Decision tree is a typical inductive approach to learn knowledge on classification. The conditions are :
- Attribute-value description: Object or case must be expressible as a fixed collection of properties or attributes having discrete values.
- Predefined classes : Categories to which examples are to be assigned must already be defined (ie supervised data).
- Discrete classes: Classes must be sharply delineated; continuous classes broken up into vague categories as "hard", "flexible", "soft".
- Sufficient data: Enough training cases to distinguish valid patterns.

Some specific applications include
◊ medical diagnosis,
◊ credit risk assessment of loan applications,
◊ equipment malfunctions by their cause,
◊ classification of plant diseases, and
◊ web search classification.

Explanation Based Learning (EBL)

Humans appear to learn quite a lot from one example. Human learning is accomplished by examining particular situations and relating them to the background knowledge in the form of known general principles. This kind of learning is called "Explanation Based Learning (EBL)".

General Approach

EBL is abstracting a general concept from a particular training example. EBL is a technique to formulate general concepts on the basis of a specific training example. EBL analyses the specific training example in terms of domain knowledge and the goal concept. The result of EBL is an explanation structure, that explains why the training example is an instance of the goal concept. The explanation-structure is then used as the basis for formulating the general concept.
Thus, EBL provides a way of generalizing a machine-generated explanation of a situation into rules that apply not only to the current situation but to similar ones as well.

Data Driven Discovery

Data driven science, in contrast to theory driven, starts with empirical data or the input-output behavior of the real system without an explicitly given theory. The modeler tries to write a computer program which generates the empirical data or input-output behavior of the system. Typically, models are produced in a generate-and-test-procedure. Generate-and-test means writing program code which tries to model the i-o-behavior of the real system first approximately and then improve as long as the i-o-behavior does not correspond to the real system. A family of such discovery models are known as BACON programs.

BACON System

Equation discovery is the area of machine learning that develops methods for automated discovery of quantitative laws, expressed in the form of equations, in collections of measured data.
BACON is pioneer among equation discovery systems. BACON is a family of algorithms for discovering scientific laws from data.
BACON.1 discovers simple numeric laws. Given a set of observed values about two variables X and Y, BACON.1 finds a function $Y = f(X)$
BACON.3 is a knowledge based system, has discovered simple empirical laws like physicists and shown its generality by Ideal gas law, Kepler's third law, Ohm's law rediscovering the and more.
The main heuristics detect constancies and trends in data, and lead to the formulation of hypotheses and the definition of theoretical terms. The program represents information at varying levels of description. The lowest levels correspond to direct observations, while the highest correspond to hypotheses that explain everything so far observed.

Clustering

Clustering is a way to form natural groupings or clusters of patterns. Clustering is often called an unsupervised learning. Clustering is one of the most utilized data mining techniques.
■ The data have no target attribute. The data set is explored to find some intrinsic structures in them.
■ Unlike Classification where the task is to learn to assign instances to predefined classes, in Clustering no predefined classification is required. The task is to learn a classification from the data.

Analogy

Learning by analogy means acquiring new knowledge about an input entity by transferring it from a known similar entity. This technique transforms the solutions of problems in one domain to the solutions of the problems in another domain by discovering analogous states and operators in the two domains.

Neural net and Genetic Learning
Are Biology-inspired AI techniques. Neural Net (NN) A neural net is an artificial representation of the human brain that tries to simulate its learning process.
An artificial neural network (ANN) is often just called a "neural network" (NN).

- Neural Networks model a brain learning by example.
- Neural networks are structures "trained" to recognize input patterns.
- Neural networks typically take a vector of input values and produce a vector of output values; inside, they train weights of "neurons".
- A Perceptron is a model of a single `trainable' neuron.

Neural network uses supervised learning, in which inputs and outputs are known and the goal is to build a representation of a function that will approximate the input to output mapping.

Genetic Learning
Genetic algorithms (GAs) are part of evolutionary computing. GA is a rapidly growing area of AI.
■ Genetic algorithms are implemented as a computer simulation, where techniques are inspired by evolutionary biology.
■ Mechanics of biological evolution:
‡ Every organism has a set of rules, describing how that organism is built, and encoded in the genes of an organism.
‡ The genes are connected together into long strings called chromosomes.
‡ Each gene represents a specific trait (feature) of the organism and has several different settings, e.g. setting for a hair color gene may be black or brown.
‡ The genes and their settings are referred as an organism's genotype.
‡ When two organisms mate they share their genes. The resultant offspring may end up having half the genes from one parent and half from the other. This process is called cross over.
‡ A gene may be mutated and expressed in the organism as a completely new trait.
Thus, Genetic Algorithms are a way of solving problems by mimicking processes the nature uses ie Selection, Crosses over, Mutation and Accepting to evolve a solution to a problem.

Reinforcement Learning
Reinforcement learning refers to a class of problems in machine learning which postulate an agent exploring an environment. The agent perceives its current state and takes actions. The environment, in return, provides a reward positive or negative. The algorithms attempt to find a policy for maximizing cumulative reward for the agent over the course of the problem. In other words, the definition of Reinforcement learning is : " A computational approach to learning whereby an agent tries to maximize the total amount of reward it receives when interacting with a complex, uncertain environment."
Reinforcement learning is “a way of programming agents by reward and punishment without needing to specify how the task is to be achieved”.

Key Features of RL
■ The learner is not told what actions to take, instead it find finds out what to do by trial-and-error search.
■ The environment is stochastic; ie., the behavior is non-deterministic means a "state" does not fully determine its next "state".
■ The reward may be delayed, so the learner may need to sacrifice short-term gains for greater long-

term gains.
■ The learner has to balance between the need to explore its environment and the need to exploit its current knowledge.

Thus the elements of Reinforcement Learning are

- Policy
- Agent’s Learning Task
- Reward
- Value Functions

Policy
It defines the learning agent's way of behaving at a given time. A mapping from states to actions (deterministic policy), or the distributions over actions (stochastic policy).
■ It is a mapping from perceived states of the environment to actions to be taken in those states.
■ It in some cases may be a simple function or lookup table; it may involve extensive computation like a search process.
■ It is the core of a reinforcement learning agent;it alone is sufficient to determine behavior.
■ In general it may be stochastic (random).

Reward Function
■ Maps each perceived state (or state-action pair) of the environment to a single number, indicating intrinsic desirability of that state.
■ Indicates what is good in an immediate sense.
■ Tells what are the good and bad events for the agent;
Example : a biological system identify rewards high with pleasure and low with pain
■ RL agent's sole objective is to maximize the total reward it receives in the long run.
■ Rewards servers a basis for altering the policy;
Example : an action selected by the policy if followed by low reward, then the policy may be changed to select some other action in that situation.
Value Functions
The value of state s under policy π is the expected return when starting from *s* and choosing actions according to π.