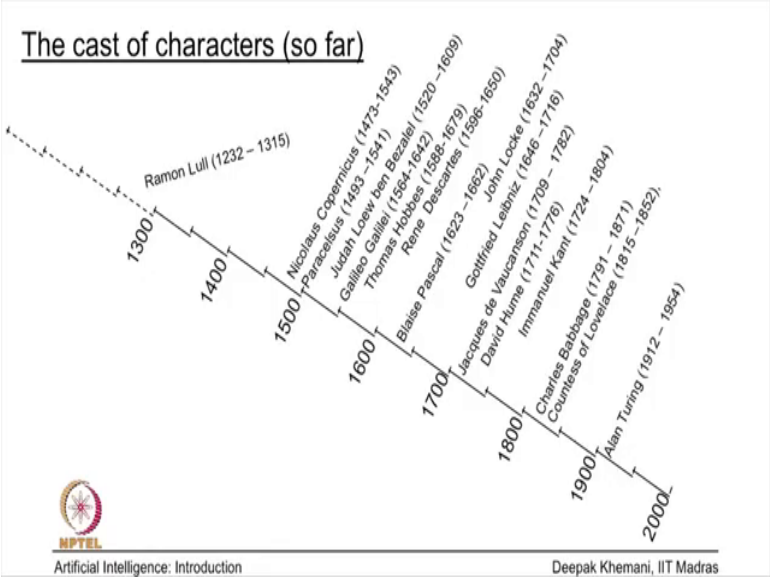
Artificial Intelligence: Search Methods for Problem Solving
Prof. Deepak Khemani
Department of Computer Science & Engineering
Indian Institute of Technology, Madras

Lecture - 12
Introduction (2013)
How did AI get its name?

(Refer Slide Time: 00:24)



So, we are entering a last leg of our introduction part and I do not know whether you can read the names on the slides, these are the people we have met so far. Starting with Nicolaus Copernicus or even before that Ramon Lull and a host of people including Galileo and Hobbes and Descartes, Pascal.

Pascal who said that perception is in our minds and if you smell, the smell of a rose then that is a reaction to the particles that are impinging upon our nose and we are sort of perceiving

based on that essentially. So, we saw that there was a strand of reasoning now this is you can say the prehistory of AI and you can see from this diagram that it is about more than 500 years of history. And, there are two strands to this and one is that the physical side of trying to make talking heads, walking statues and statues which can you know nod their heads and that kind of stuff.

The engineering or the physical or the contraption side of it because there was this belief that it can move autonomously, it must be able to think also essentially. So, it is a leap of faith that one use to make. The other strand was the emergence of the notion of the mind, at some point you know creatures like us would simply live in a world and see the world and believe that what we see is what the world is like.

But, then along came somebody like Copernicus and he said that you think that the sun is going round the earth during the day, but that is not what is happening. What is happening is that the earth is rotating and it creates an illusion of the sun going round the earth. So, the fact that what you see is not necessarily what is out there had already started coming out. And, gradually then the distinction between what we see and what we think started happening and at some point Descartes said there are two worlds out there.

One is the world of our mind and the other is a world of the body and he had this idea of mind body dualism. Then as we moved along we saw Kant for example, Immanuel Kant one of the most influential philosophers from Europe who said that we perceive the world in terms of a priori knowledge that we have in our heads. And, we mold the world that we see into those a priori knowledge structures that we have.

Of course, he did not use the term knowledge structures which we use nowadays or concepts that we have essentially. And, at the same time the mechanical contraptions were becoming more and more sophisticated, there was this duck in France of Vaucanson. So, if some of you saw BBC every Sunday if you see BBC, you get something for this course.

So, this last Sunday BBC showed a news item in which in the South of France they have opened the museum of all these talking, walking mechanical creatures essentially which

apparently were very popular. And, they used to keep them in shop windows to attract shoppers and things like that and I could see there this Vaucanson's duck also amongst the displays.

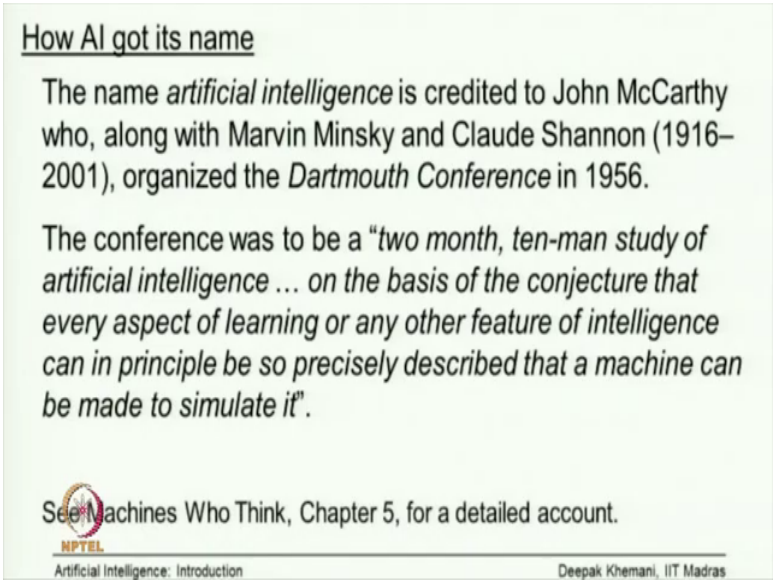
So, they were getting sophisticated and we see that from; so, these two strands are merging together, you know this moving creatures and thinking creatures in some sense. So, Pascal for example, we will sort of recognize him here for the fact that he was the first person to invent a calculating machine. Of course, it could only do addition, but nevertheless it was a calculating machine which was sort of improved later by Leibniz into something which could do more than addition it could do multiplication and so on.

And, it became more and more sophisticated till we came to Charles Babbage who invented a machine which could store a program and run the instructions in that program which is the notion of computers, that we are still working with essentially. We also met Alan Turing who sort of tried to put down this debate on what is intelligence, can machine thinks and that kind of a stuff he proposed the Turing test that we saw earlier. He did many other things and we will just have a brief mention on of him later essentially.

So, let me remind you of this definition by Haugeland, AI is a quest for building machines with minds of their own and we had asked this question towards the end of the last lectures. So, what is what are minds? We will come to that question later today, but a little bit later; before that we let us complete the history. So, we have seen the pre-digital computer eras so far; the mechanical contraptions the people used to build.

How did AI progress after the digital computer came into being which is just around the time when Alan Turing was around.

(Refer Slide Time: 05:36)



So, let us first begin with this piece of information which tells us; how did we get this name artificial intelligence essentially ok. And, the name was devised by John McCarthy, I think we all know John McCarthy in for some form of another amongst other things he invented the language Lisp which became very popular in AI for many years.

So the name is credited to John McCarthy and Marvin Minsky along with Claude Shannon, who organized this conference called the Dartmouth Conference in Dartmouth college in 1956, where McCarthy is credited with having devices named artificial intelligence. Now, many people have said that no this name is not a nice name, you should use something like heuristic programming or machine intelligence or something else. But, somehow the name is stuck since that time and we all know this area as artificial intelligence.

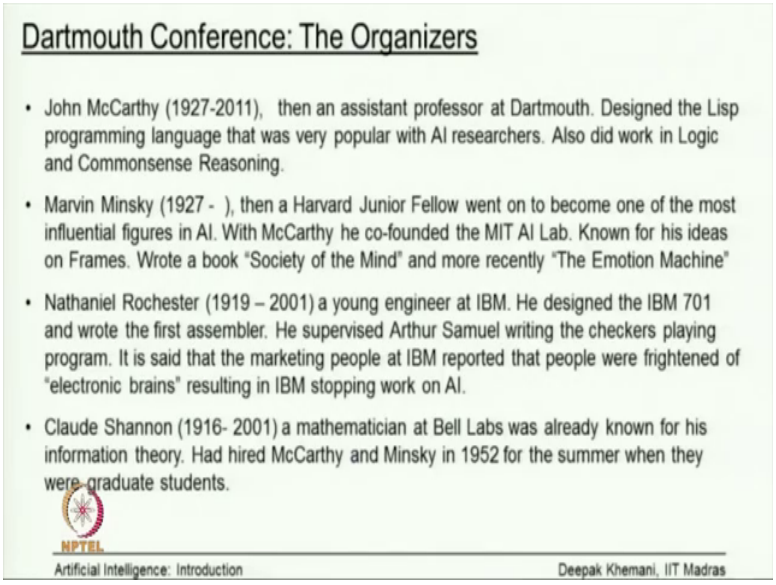
So, Haugeland for example, suggest that you could call it synthetic intelligence. So, artificial is and he makes this comparison with pearls for example. Now you have real pearls, you have artificial pearls which are kind of fake essentially, but you also have synthetic pearls which are not fake, but which are pearls, but which have been sort of made by humans.

So, the conference was organized and its charter was that it would do a two month, ten-man study of artificial intelligence, a name was coined there on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it ok. So, the focus or the emphasis is that when we are talking about intelligent behavior it is something which can we can describe up to the minutest detail.

And, if you can do that we can make a machine do it essentially; so, that was the idea behind that. So, who are the people who organize this? We will see them in a moment. So, for those of you who are interested in history; so, we you should look at these two books which I mentioned earlier; in this one book called Machines Who Think that is shown here by Pamela McCorduck. The Chapter 5 describes the full chapter on the Dartmouth conference.

And, the other book is John Haugeland which is AI - The Very Idea which is a more philosophical side of things essentially; we will come back to that a bit later.

(Refer Slide Time: 08:15)



So, who are the organizers of this conference? We have already said that they were John McCarthy; he was then an assistant professor at Dartmouth and he has done many. So, these people who have contributed so much to AI in one way or the other that it is you know not easy to list there what all they did, they were very active people.

So, McCarthy invented Lisp, he invented something called situational calculus which we will may or may not see in this course. He is also credited with having invented the alpha beta algorithm which we will see a bit later in this course. And, he did lot of work on logic and common sense reasoning essentially; which we will see if we can come to that later.

Marvin Minsky the only person of this who is still alive was in a junior fellow at Harvard and he and McCarthy towards set up the MIT lab, AI lab in MIT. And, as you will see in the

history part today most of this work in AI was concentrated in a few places in the US and a few places in Europe and it is not as if everybody everywhere was working on AI.

So, MIT lab was one, CMU was another place, Stanford was another place and you know a couple of places in Europe essentially. So, Minsky is very well known for his idea of frames which is a way of structuring knowledge into interconnected components. And, it is basically the forerunner of what we called as object oriented programming nowadays essentially.

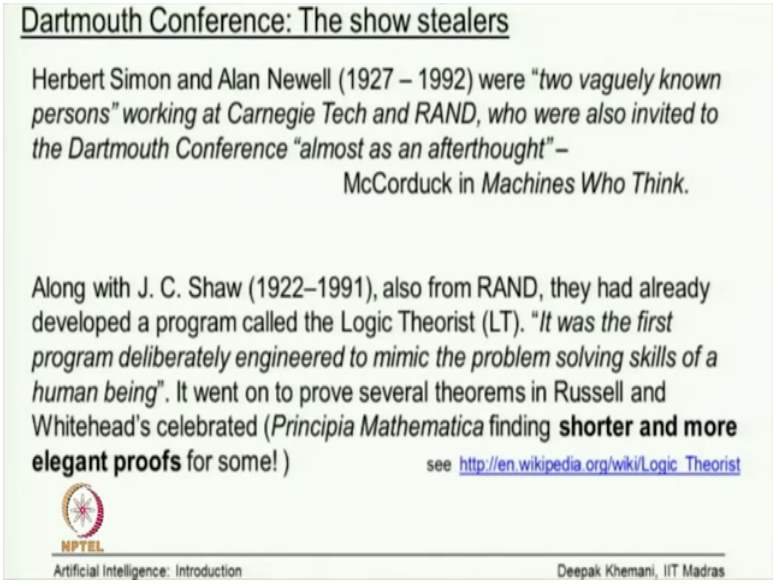
He also wrote a very influential book called “Society of the Mind” and more recently a book called “The Emotion Machine” essentially. Somebody had once pointed out during our initial lectures that machines cannot display emotions. So, maybe you should look at that book. Another person there was Rochester who was the inventor or the designer of this IBM 701 machine which was the best machine around at that time.

He wrote the first assembler for that machine. He supervised Arthur Samuel into writing a program for playing the game of checkers which we will talk about a little bit which was one of the early successes of AI essentially. Now, it turns out that this Samuels program was a learning program and Samuels goals were actually learning. He wanted to see how computers could learn and his program was a program which became better and better as it played more and more essentially.

And, this kind of a generated a kind of a fear amongst the people that these machines will become smarter than us, more powerful than us and things like that, we will come back to that point little bit later when we talk about this checkers program. And finally, Claude Shannon everybody knows Claude Shannon because of his information theory and in fact, he was the person who had hired Minsky and McCarthy as interns when they were graduate students.

And, it is there that they got this idea of putting together this conference which will talk about this new field which was coming up called artificial intelligence.

(Refer Slide Time: 11:33)



But, there were a couple of guys who were in some sense a show stealers at that conference essentially. So, let us first see who they were; the names we have might have mentioned before Herbert Simon and Allen Newell. And, Pamela McCorduck says that they that about them the two vaguely known persons working at Carnegie tech at that time CME was not the Carnegie Mellon University, it later on became CMU, but at that point it was Carnegie Tech and RAND, who were also invited to the Dartmouth conference and as she writes almost as an afterthought.

And, it is these two people who really created an big impact at the conference because they had along with J. C. Shaw who also worked at RAND, build this written this program called the Logic Theorist. It was a logical reasoning machine it or theorem proving machine, it could prove theorems in mathematics; LT as short it was. So, they say about this it was a first

program deliberately engineered to mimic the problem solving skills of a human being essentially.

So, Simon and Newell were greatly influenced by the way human beings solve problem because after all we are sort of existential examples of smart creatures. You know we sort of are thinking creatures, if you do not want to call us machines who operate very effectively in the world you know solving problems and getting along and so on.

And, he wrote a book they wrote a book called human problem solving which became very influential later essentially. Now, this program logic theorist went on to prove several theorems from Russell and Whitehead. See Russell and Whitehead had embarked upon this grand exercise of formalizing all knowledge. And, they said you know everything that you can do in mathematics will put it down in piece of paper, their great dream was shattered in 1931 by.

Student: Kurt Gödel.

Kurt Gödel, when he came and proved that you cannot become powerful you cannot construct powerful enough systems, reasoning systems which are consistent at the same time. So, either you can be very powerful in the sense very expressive that you can talk about all kinds of things or you can be consistent, but not both at the same time. And, he showed that this is something which will always follow, if you try to build powerful enough systems.

And, as some of you might know his arguments are basically centered around self reference and self negating sentences. So, sentences like I am lying or the story about this barber which Russell and Russell was so worried about; that if there is a village in which the rule is that everyone who does not shave himself is shaved by the barber; then the question is who shaves the barber essentially?

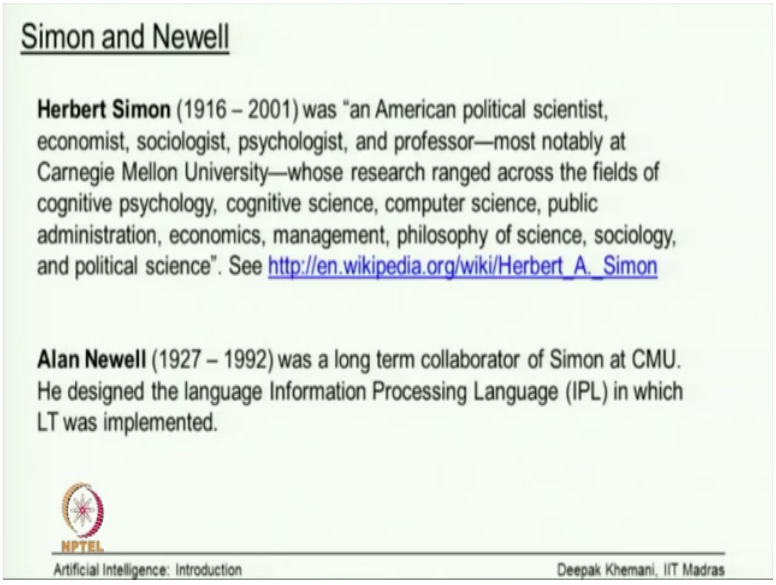
Because, the barber if he shaves himself then he is shaving himself and therefore, he cannot shave himself. So, some this kind of conundrums come through self referential sentences. And, Kurt Gödel showed that any formal system which is expressive enough will end up becoming

inconsistent which means in our kind of things that we want to study in that higher order logics are never going to be consistent and complete at the same time. We will come to these notions at some later point of time.

But, the simpler logics as first order logic or predicate logic is good enough for us and first order logic can be seen to capture everything that we are doing in programming essentially. So, our programs can whatever we can express in programs we can do consistently in some sense. Now, this program LT produced some shorter and more elegant proofs that were present in this Principia Mathematica; this is a book by Russell and Whitehead.

And, apparently so, this story goes I do not know whether it is true or not, but apparently the Journal of Symbolic Logic or journal of logic or something refused to accept the paper because it was authored by a computer program, co-authored by a computer program. I do not know whether the story is true or not, but you can find it in some places essentially.

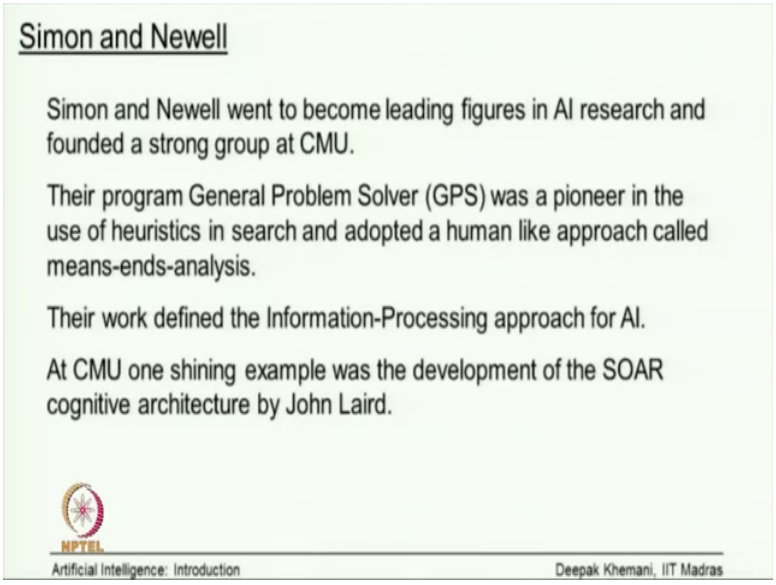
(Refer Slide Time: 16:02)



So, let us first talk a little bit about Simon and Newell. Simon was a multifaceted person as you can see from this quote from Wikipedia; he was a political scientist, economists, sociologists, psychologists and professor mostly at CMU whose research ranged across all these fields cognitive psychology, cognitive science, computer science, public administration, economics and so on and so forth.

He went on to get a Nobel Prize in economics and his longtime associate was Allen Newell about 10 years his junior and they did a lot of collaborative work together essentially. So, Allen Newell created this language called IPL in which LT was implemented.

(Refer Slide Time: 16:41)



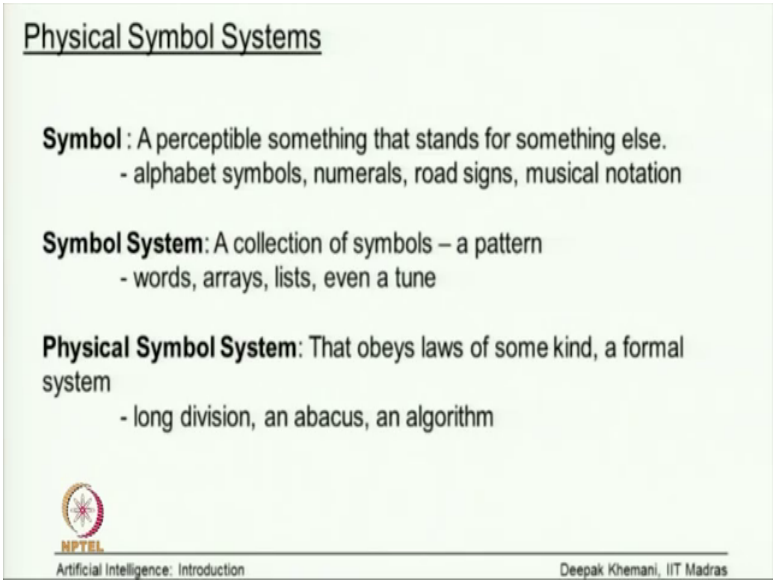
Some a little bit more on them because they give us something which we base our work on. So, they became leading figures at CMU and they wrote this program called General Problem Solver which was based on human problem solving and how human beings use heuristics to solve problem. And, we will visit this general problem solve solver idea, this idea of means ends analysis which is a heuristic that we use; we will see that sometime later in the course essentially.

Their work also brought to focus the information processing approach to AI which means that you are talking about, that if you want to create intelligent systems, it is rough to information processing. As opposed to this others standoff effort which was to say that we will build systems from bottom up.

We will put together a components which make intelligent systems and so on and eventually they will become intelligent. He said you do not have to do all that, you can work at the information level or as some people call as a knowledge level and build intelligent systems essentially.

And, one of the things which came out of CMU, one of the many things which came out of CMU was this cognitive architecture called SOAR which you can even now download and use to build good applications. So, one of the things they talked about was this architecture for AI, what do you need for AI.

(Refer Slide Time: 18:10)



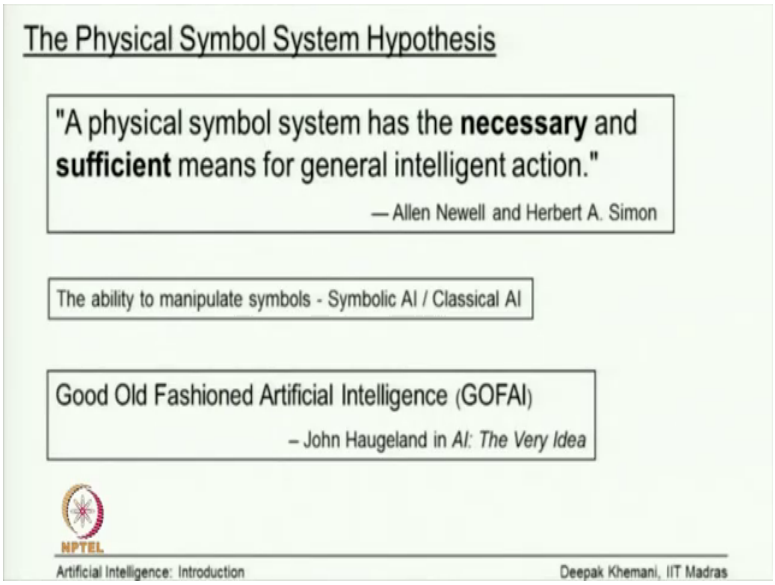
So, you have this idea of the physical symbol systems and the symbol is something as far as we are concerned; a perceptible something which stands for something else. So, a symbol stands for something else essentially. If you write the numeral 7, it stands for the number 7. Of

course, it is not the number 7, it just stands for the number 7 and we could have in a different script we could have written it differently.

A symbol system is a collection of symbols. So, for example, a data structure or a English language word or even a musical tune essentially. So, you put them together you have a symbol system. So, you have an alphabet which is made of symbols and then you put together things of from that alphabet and you have a symbol system.

And, a physical symbol system is something which obeys laws which are like the laws of physics essentially ok. So, in some sense if they if you can manipulate them using well defined laws or rules then they are physical in that sense they are physical ok, in the sense that they can be manipulated according to this law. So, anything you can use algorithms or the procedure for long division for example, and so on and so forth.

(Refer Slide Time: 19:15)



The important statement that they made is known as a physical symbol system hypothesis. It says that a physical symbol system has a necessary and sufficient means to generate intelligent action. So, what they are saying that all you need in our terminology all you need to build intelligent systems is the ability to create data structures and write algorithms which will operate upon those data structures.

You need nothing else essentially, that is the basic infrastructure you need. So, unlike for example, Roger Penrose who feels that the human mind or human brain has some kind of physics which is going on which we cannot replicate; they said nothing of the sort. If you can do information processing which means if you can operate on symbol systems using well defined algorithms, you can create intelligent behavior.

So, this is known as a symbolic AI or classical AI; classical AI follows this principle that it is a top down design approach to building intelligent systems; that you will create your data structures and you will write your algorithms and you will produce your intelligent systems, whereas, Haugeland calls it good old fashioned AI.

(Refer Slide Time: 20:29)

Next

The Chess Saga