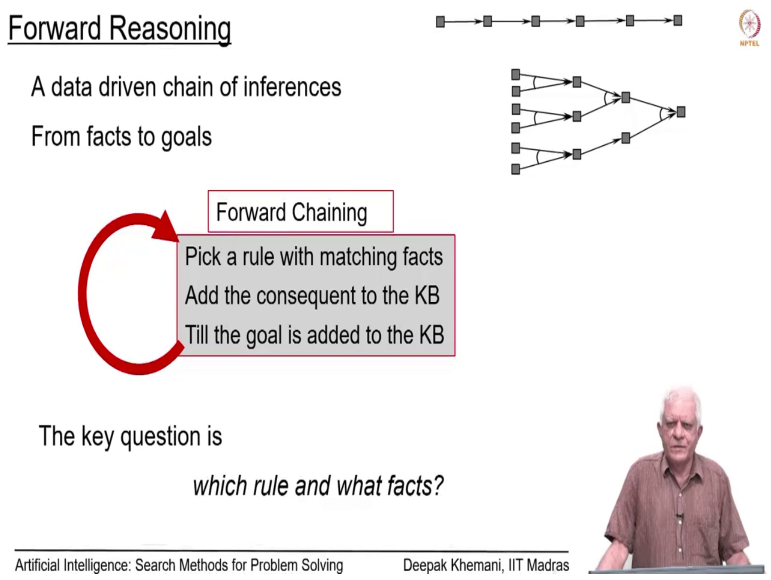
Artificial Intelligence: Search Methods for Problem Solving
Prof. Deepak Khemani
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Chapter – 11
A First Course in Artificial Intelligence
Lecture – 82
Deduction as Search - Forward Reasoning

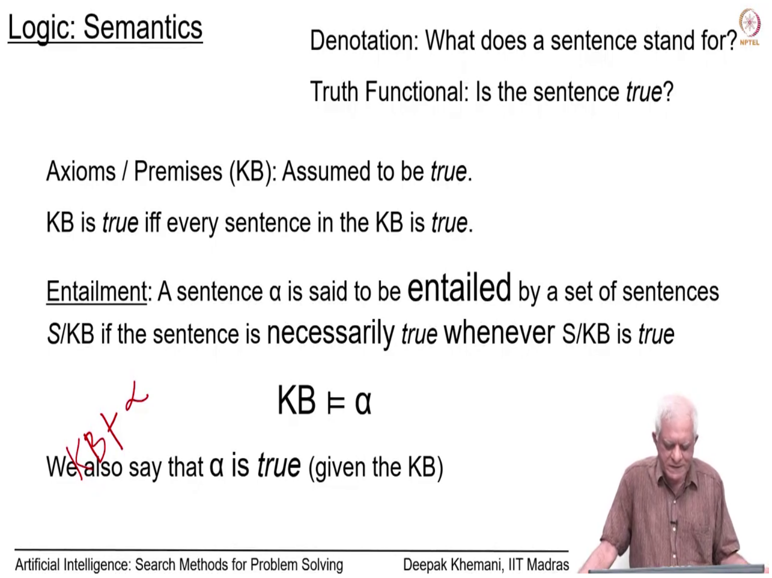
(Refer Slide Time: 00:13)



So, welcome back in our quest for finding a theorem prover or an algorithm which will prove theorems for us. Mathematicians use the word theorem for any true statement, which is true. So, you have some goal and you want to show that this goal is true. We say that we are doing theorem proving essentially.

And we have seen that the basic idea is going to be one of the algorithms that we are going to be looking at is forward reasoning or forward chaining, where we chain the rules from the fact towards a goal and we keep doing that till the goal is added to the knowledge base.

(Refer Slide Time: 00:52)



Let us talk about the fact that we want to do meaningful things. We want to do use logic to help us arrive at true statements. How do we convince ourselves that our machinery let us call it machinery, it is a program algorithm which is working on some representation and the logic machine let us call it that what it is producing are true statements.

So, we need some kind of a theoretical foundation justification for that essentially and that takes us into the world of semantics. I had said that you know meaning its only lies in the mind of the beholder that the symbol system only does manipulation of symbols.

So, there are some questions which as users of this logic machine we will want to ask as to the one is; there are two aspects to semantics. One is denotation, what does the sentence stand for what is it saying essentially. The other is truth functional, is that sentence true or not true essentially.

So, logic of course, is more concerned with truth functional semantics and so, we are interested in producing only true statements that is what our primary concern is going to be. What it stands for is going to be still largely left to us essentially.

So, we have said that axiom axioms or premises are assumed to be true or given to be true. So, we just accept them without questioning. A knowledge base is true, if every sentence in the knowledge base is true that was kind of simple and logical. We have the notion of entailment. We said earlier that is the consequent necessarily true that another word for that is entailment.

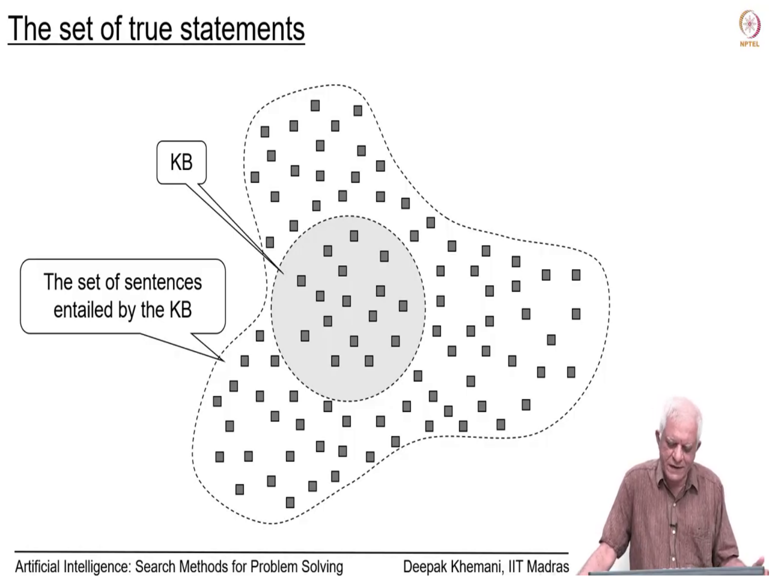
We say that a sentence is said to be entailed by a set of sentences S or the knowledge base, whatever you want to call them; if the sentence is necessarily true, whenever the knowledge base is true. So, give us a true knowledge base. Then we say alpha is entailed. If whenever the knowledge base is true, alpha must necessarily be true. Again you must distinguish between the fact of provable versus entailment.

Provable says that we have a set of rules by repeated application of which you can produce the new sentence. Entailment says that this sentence alpha is necessarily true whenever the knowledge base is true. So, the two notions are different say. Entailment is a semantic notion; proof or provability is a syntactic notion.

We use this symbol KB entails alpha to say that that whenever the knowledge base is true, alpha will be true. Contrast this with the other statement that I had mentioned some time ago let KB proves alpha.

So, alpha is provable given the knowledge base. These two are different things even though of course, they are connected to each other in some way as we will see. We also say that alpha is true given the knowledge base essentially or alpha is entailed which are equivalent statements.

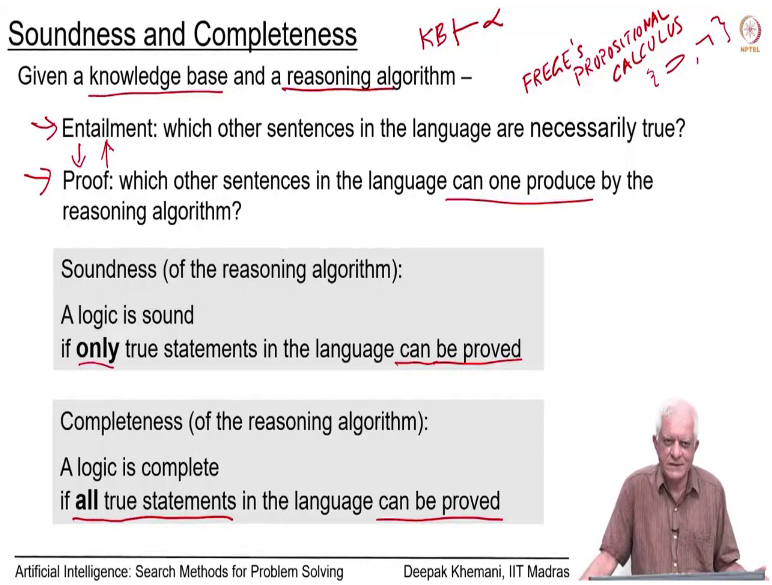
(Refer Slide Time: 04:23)



So, given a knowledge base, we will have a set of true statements as shown by this curvy figure around the knowledge base that that there are some sentences which will be true as a consequence of the knowledge base which is true.

So, in the beginning of this set of videos we said that you know Sneha is the mother of Snigdha and as a consequence of that we can say Snigdha is the daughter of Sneha or if it was the otherwise around I do not remember, but you can make those kinds of things which are entailed essentially. So, given a knowledge base there is a set of statements which is entailed.

(Refer Slide Time: 05:03)



Now, when can we trust our logic machine? So, there are two properties that we talk about in terms of logic machines. One is called soundness and one is called completeness essentially. So, given a knowledge base and a reasoning algorithm. What is the reasoning algorithm? The reasoning algorithm is the procedure we are using for adding new statements.

So, we have started by looking at forward chaining, we will also look at backward chaining or backward reasoning in this week. But, there are other approaches as well that unfortunately we do not have time to get into. But, that is a reasoning algorithm as to how do you add new formulas to the system or how do you prove that a sentence is entailed.

So, reasoning, when we say reasoning we are concerned with this relation. Can we prove alpha? So, given a knowledge base. So, we have given two things and a reasoning algorithm. Entailment is simply a property of the knowledge base. It says that which other sentences are necessarily true ok. We are not talking about reasoning here.

We are saying we are talking about facts that you give us a knowledge base which is true and we at least theoretically or conceptually in principle we can define a set of sentences, which are necessarily true. The proof on the other stand is concerned with the reasoning algorithm.

It says that given a knowledge base, which other sentences in the language can one produce by the reasoning algorithm which is obviously, a different thing essentially. It is not talking about truth at all. It is simply saying that you have given me a rule set of rules of inference and you have given me a knowledge base, I can apply all the rules of inference and can I produce this alpha or not essentially. So, reasoning as I said is concerned with provability essentially.

Now, logic has been around for a few hundred years essentially and lots of logicians have experimented with different kinds of rules of inference, different kinds of operators connectives to use in the language. So, we have we have seen you know things like implication AND, OR, NOT and so on.

There are 16 binary connectives. People have looked at a subsets of these. So, for example, if you look up Frege, Frege's propositional calculus and there is a nice Wikipedia page about this. It uses only the implication and the negation sign and these are the only two the only two connectives it uses.

And Frege showed that this was enough to do all kind of reasoning, logical reasoning that you want do. And when I say enough it basically means brings the notion of completeness which we are just about to define. So, do look up Freges propositional calculus, but a lot of time people have spent looking at which subset of operators we should use, which rules of inference we should use and so on and so forth.

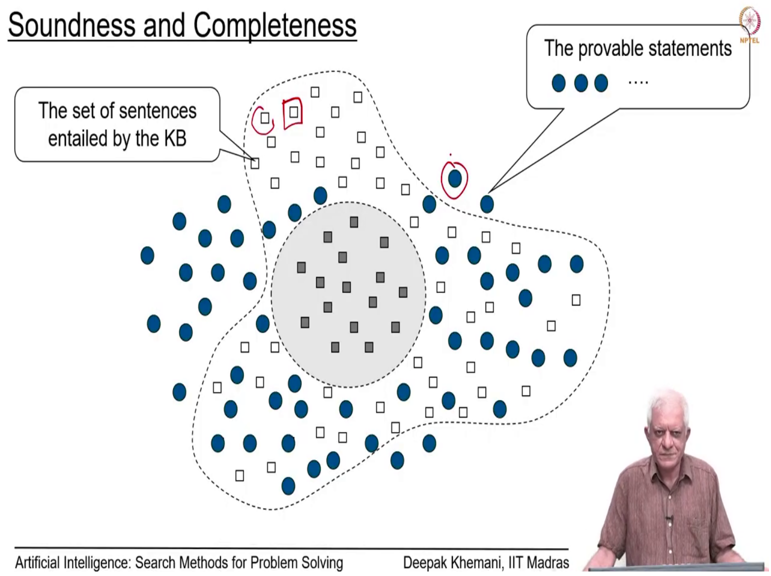
So, soundness that is necessary because we want logic to find true statements for us and we say that our logic is sound, if only true statements in the logic can be proved. So, again remember we are talking about the reasoning algorithm we are talking about which can be proved essentially.

But, the key thing is that you should be able to produce only true statements, which means that if the logic machine gives you a statement you are happy to accept it as a true statements because you know that it will produce only true statements.

The other property of completeness says that all true statements can be proved. What do we mean by all true statements? That essentially if a statement is entailed by the knowledge base, it must have a proof in this logic machine that we are talking about. The soundness says that if the statement has a proof, it must be entailed. So, as you can see soundness and completeness relate entailment and proof to each other.

So, entailment is here proof is here and if you go from entailment to proof then we say it is complete. If we go from proof to entailment then we say it is sound. If it is provable it must be true. We had briefly mentioned that soundness has to do with soundness of rules. If a rule of inferences sound then the whole chain of inference would be sound you see, but we will not get too much into that essentially here.

(Refer Slide Time: 10:52)



So, this is a notion of soundness and completeness in some sense. We saw that these squares that you see here like these, these are the set of entail statements. We saw that given a knowledge base there is a set of statements which are entailed. At the same time given a knowledge base there are sets of statements which can be proven by this logic machine that we are talking about. Sound, if a logic is to be sound and complete these two sets must be identical.

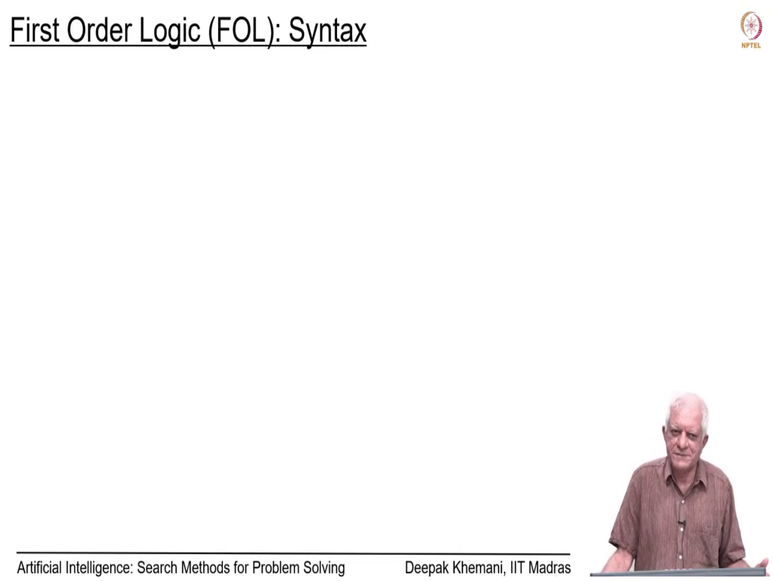
If a logic is to be sound, remember that soundness says that if there is a proof, then it is entailed. Then the set of provable statements must be a subset of the entail statements. If a logic is complete and what does completeness say? It says that if it is true you will find a proof for it. So, the set of true statements or the set of entail statements is a subset of the set of provable statements.

But, if it has to be sound and complete the two sets must be equal essentially. So, soundness and completeness are necessary properties that we want if we want to use the logic machinery. And Godel showed almost a century ago that first order logic is both sound and complete. What that means, is that you can devise logic machines in first order logic which will be complete and sound essentially.

Godel also showed that second order logic can never be sound incomplete. Now, that is an intriguing statement trying because he makes a statement of something which can never happen that you can never devise a second logic system which will be always sound and always complete it cannot be both at the same time.

And that is the incompleteness theorem many people have used this or some people have used this as an argument to say that machines can never be intelligent, but the connection is very tenuous essentially.

(Refer Slide Time: 13:11)



So, let us now focus on first order logic and also start looking at proof systems. So, let me take another short break while you digest the notion of soundness and completeness. And now, we will get down to the nitty gritty of representation of reasoning in first order logic essentially, but we will also look at some examples from propositional logic because they are simpler to depict essentially. So, we will do that in the next segment.